Santosh Kumar Yadav

Discrete Mathematics with Graph Theory

Ane Books Pvt. Ltd.

Springer

Discrete Mathematics with Graph Theory

Santosh Kumar Yadav

Discrete Mathematics with Graph Theory

Ane Books Pvt. Ltd.

Springer

Santosh Kumar Yadav
Shri Jagdishprasad Jhabarmal Tibrewala
University
Rajasthan, India

ISBN 978-3-031-21320-5 ISBN 978-3-031-21321-2 (eBook)
https://doi.org/10.1007/978-3-031-21321-2

Jointly published with Ane Books Pvt. Ltd.
In addition to this printed edition, there is a local printed edition of this work available via Ane Books in South Asia (India, Pakistan, Sri Lanka, Bangladesh, Nepal and Bhutan) and Africa (all countries in the African subcontinent).
ISBN of the Co-Publisher's edition: 978-93-82127-18-5

© The Author(s) 2023

This work is subject to copyright. All rights are solely and exclusively licensed by the Publisher, whether the whole or part of the material is concerned, specifically the rights of reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publishers, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publishers nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publishers remain neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

As long as a branch of science offers an abundance of problems, so long it is alive; a lack of problem foreshadows extinction or the cessation of independent development. Just as any human undertaking pursues certain objects, so also mathematical research and studies require its problems. It is by the solution of problems that the investigator tests the temper of his steel; he finds new methods and new outlooks, and gains a wider and freer horizon.

Almost everybody asks, what is discrete mathematics? It is the mathematics of discrete (distinct and disconnected) objects. Frankly speaking, it is the study of discrete objects and relationships that bind them. An analogous situation exists in the field of computers. Just as mathematics can be divided into discrete and continuous mathematics, computers can be divided into digital and analog. Digital computers process the discrete objects 0 and 1, whereas analog computers process continuous data that is, data obtained through measurement. Thus, the terms discrete and continuous are analogous to the terms digital and analog, respectively.

This book is the fruit of many years of many dreams; it is the product of my fascination for the myriad applications of discrete mathematics to a variety of courses such as Data Structures, Algorithmic Analysis, Combinatorics, Number Theoretic Structures, Knowledge Spaces and Graph Theoretic Problems.

The book has been designed to meet the requirement of the students of undergraduate and postgraduate level in the field of Computer Science, Information Technology, Mathematical sciences and Physical Sciences. No formal prerequisites are needed to understand the material or to employ its power, except a very strong background in college algebra. The text contains in depth coverage of all major topics proposed by professional associations/universities for a discrete mathematics course. It emphasizes problem-solving techniques, pattern recognition, conjecturing, induction,

applications of varying nature, proof technique, algorithmic development, algorithm correctness and numeric computations. A sufficient amount of theory is included for those who enjoy the beauty in development of the subject, and a wealth of applications as well as for those who enjoy the power of problem solving techniques.

Biographical sketches of about 25 mathematicians and computer scientists who have played a significant role in the development of the field are threaded into the text to provide a human dimension and attach a human face to major discoveries. Each section in the book contains a generous selection of carefully tailored examples to classify and illuminate various concepts and facts. Theorems are the backbones of mathematics. Consequently, this book contains the various proof techniques, explained and illustrated in details. Most of the concepts, definitions, and theorems in the book are illustrated with appropriate examples. Proofs shed additional light on the topic and enable students to sharpen thin problem-solving skills. Each chapter ends with a summary of important vocabulary, formulae, properties developed in the chapter and list of selected references for further exploration and enrichment.

Dr. Santosh Kumar Yadav

Acknowledgments

A number of people, including my fellow professors, research scholars and students have played a major role in substantially improving the quality of the manuscript through its development. I am truly grateful to each one of them for their unfailing encouragement, cooperation and support.

To begin with, I am sincerely indebted to the following for their unblemished enthusiasm and constructive suggestions:

Dr. Satish Verma, Mr. Dharmendra Kumar, Dr. Anju Gupta, Dr. Sudha Jain, Dr. D.K. Paliwal, Dr. Kanta Khurana, Dr. Preeti Upreti, Dr. Shashi Tiwari, Mr. Sushil Kumar, Mr. Sunit Kumar, Mrs. Anupam Goel, Dr. Durgesh Kumar, Mrs. Malti Kapoor, Mrs. Ritu katuria, Ms. Shipra Gupta, Dr. Pankaj Jain, Dr. L. Bandopadhyaya, Dr. Pramod Arora, Ms. Gunjan Khurana, Mrs. Dolly Jain, Dr. Arvind Kumar, Dr. Preeti Dharmara, Dr. S.K. Sahdev, Dr. Pramod Kumar, Dr. Pradumn Kumar, Dr. Neeraja Chawla, Dr. Rama Verma, Mrs. Usha Kiran Sekri, Dr. Dhanpal Singh, Mr. Brij Mohan, Mr. Amit Kumar, Dr. Asha Gauri Shankar, Dr. Sudha Gupta, Dr. Neera Bohadur. Mrs. Pragati Gautam, Dr. Reeta Malhotra, Dr. Ravi Prakash, Mr. Susil Kumar Azad, Dr. Shobha Bhagai, Mrs. Nandita Narayan, Dr. Seema Kurana, Dr. Pawan Bala Gupta, Dr. Naveen Kumar, Dr. Sunil Kumar Muttoo (All from Delhi University) Dr. M.K. Gupta, C.C.S. University, Meerut. Dr. S.C. Aggarwal, Shobhit University Meerut. Dr. A.K. Dwivedi, V.B.S. Purvanchal University, Jaunpur. Dr. M.N. Hoda, Bharti Vidyapeeth, New Delhi. Dr. Devender Tayal, G.G.S. I.P. University, New Delhi. Dr. Sanjeev Kumar, Dr. B.R. Ambedkar University, Agra. Dr. Romesh Kumar, Jammu University, Jammu. Dr. Satyaveer Singh, J.J.T. University, Rajasthan. Dr. K.Perumal and Dr. Kalaimani, M.K. University, Madurai. Dr. Vinod Kumar, Kurukshetra University Kurukshetra, Dr. A.P. Singh, Central University Rajasthan, Dr. C.K. Goel and Dr. Vandani Verma, Amity University, Noida. Mr. Prashant Kushwaha, Banasthali Vidyapeeth, Rajasthan. Dr. Arjun Singh and Dr. P.L. Yadav of Govt. P.G. College,

Gurgaon. Mr. Manoj Kumar, NBGSM College, Sohna Gurgaon. Mr. Alok Kapil, Ravi Kumar, FIT, Meerut. Mr. Pankaj Kumar Gupta, G.B.I.T. Meerut and many more.

I would fail in my duty if I do not thank my publisher Mr. J.R. Kapoor for providing me full liberty of my views and good designers. I am also grateful to my family members especially my wife Seema, kids Ayan and Akshita for proof reading.

Lastly I want to thank my readers who have always encouraged me and appreciated my work. I must accept responsibility, for any errors that may still remain. I would certainly appreciate receiving comments about any unwelcome surprises, alternate or better solutions, exercises, puzzles and applications.

Dr. Santosh Kumar Yadav

Contents

0. PRELIMINARIES **1–14**

0.1 Numbers 1
0.2 Euclid's Algorithm 3
0.3 Fundamental Theorem of Arithmetic 4
0.4 Euclid's Theorem 6
0.5 Congruence Modulo ***m*** 6
0.6 Chinese Remainder Theorem 7
0.7 Fermat's and Euler's Theorems 9
0.8 Exponents and Logarithms 10
0.9 Sums 11
0.10 Mapping 12
Suggested Readings 14

1. THE LANGUAGE OF SETS **15–66**

1.1 Introduction 15
1.2 Elements and Notations of Sets 16
1.3 Construction of Sets 17
1.4 Types of Sets 19
1.5 Set Operations 25
1.5.1 Intersection of Sets 25
1.5.2 Union of Sets 26
1.5.3 Disjoint Set (Mutually Exclusive) 27
1.5.4 Ordinary Difference of Sets ($A - B$) 27
1.5.5 Complementation of Sets 27

1.5.6 Universal Set and its Complement 27
1.5.7 Symmetric Difference (Boolean Sum) 28
1.6 Venn Diagrams 28
1.7 Some Basic Results 32
1.8 Properties of Set Operations 34
1.8.1 Properties of Intersection on Sets 34
1.8.2 Properties of Union of Sets 35
1.8.3 Number of Elements in a Union of two or more Sets 39
1.9 De-Morgan's Laws 40
1.10 General form of Principle of Inclusion and Exclusion 44
1.11 Laws of Sets 63
Summary 63
Suggested Readings 65

2. BASIC COMBINATORICS 67–114
2.1 Introduction 67
2.2 Basic Counting Principles 68
2.2.1 The Principle of Disjunctive Counting (Sum Rule) 68
2.2.2 The Principle of Sequential Counting (Product Rule) 69
2.3 Factorial 71
2.4 Permutation and Combination 73
2.4.1 Cyclic Permutation 76
2.4.2 Pascal's Identity 76
2.4.3 Vandermonde's Identity 77
2.4.4 Pigeonhole Principle 78
2.4.5 Inclusion–Exclusion Principle 79
2.5 The Binomial Theorem 93
2.6 *n*th Catalan Number 95
2.7 Principle of Mathematical Induction (P.M.I) 96
2.8 Recurrence Relations 99
Summary 110
Suggested Readings 113

3. MATHEMATICAL LOGIC 115–180

3.1 Introduction 115
3.2 Propositions (Statements) 117
3.3 Connectives 117
3.3.1 Negation 118
3.3.2 Conjunction 119
3.3.3 Disjunction 119
3.3.4 Conditional 120
3.3.5 Biconditional 120
3.4 Equivalence of Formulae 121
3.5 Well-Formed Formulae (WFF) 122
3.6 Tautologies 122
3.7 Principle of Duality 123
3.8 Two State Devices 128
3.9 The Relay-Switching Devices 129
3.10 Logic Gates and Modules 130
3.10.1 OR, AND and NOT Gates 130
3.10.2 Two-Level Networks 132
3.10.3 NOR and NAND Gates 132
3.11 Normal Forms (Decision Problems) 141
3.11.1 Disjunctive Normal Form (DNF) 141
3.11.2 Conjunctive Normal Form (CNF) 145
3.11.3 Principal Disjunctive Normal Form (PDNF) 146
3.11.4 Principal Conjuctive Normal Forms (PCNF) 148
3.12 Rules of Inference 151
3.13 Automatic Proving System (Theorems) 152
3.14 The Predicate Calculus 164
3.14.1 Statement Functions, Variables and Quantifiers 166
3.14.2 Free and Bound Variables 166
3.14.3 Special Valid Formulae using Quantifiers 167
3.14.4 Theory of Inference for the Predicate Calculus 168
3.14.5 Formulae Involving More than one Quantifier 169
Summary 175
Suggested Readings 179

4. RELATIONS 181–236

4.1 Introduction 181
4.2 Product Sets 182
4.3 Partitions 182
4.4 Relations 183
4.5 Binary Relations in a Set 183
4.6 Domain and Range of a Relation 184
4.6.1 Number of Distinct Relation From set *A* to *B* 185
4.6.2 Solution sets and Graph of Relations 189
4.6.3 Relation as Sets of Ordered Pairs 190
4.7 The Matrix of a Relation and Digraphs 190
4.8 Paths in Relations and Digraphs 191
4.9 Boolean Matrices 194
4.9.1 Boolean Operations AND and OR 195
4.9.2 Joint and Meet 195
4.9.3 Boolean Product 195
4.9.4 Boolean Power of a Boolean Matrix 195
4.10 Adjacency Matrix of a Relation 198
4.11 Gray Code 198
4.12 Properties of Relations 200
4.12.1 Reflexive and Irreflexive Relations 201
4.12.2 Symmetric, Asymmetric and Antisymmetric Relations 201
4.12.3 Transitive Relation 202
4.13 Equivalence Relations 205
4.14 Closure of Relations 207
4.15 Manipulation and Composition of Relations 208
4.16 Warshall's Algorithm 216
4.17 Partial Order Relation 225
4.17.1 Totally Ordered Set 226
4.17.2 Lexicographic Order 226
4.17.3 Hasse Diagrams 228
Summary 230
Suggested Readings 235

5. FUNCTIONS 237–270

5.1 Introduction 238
5.1.1 Sum and Product of Functions 239
5.2 Special Types of Functions 242
5.2.1 Polynomial Function 244
5.2.2 Exponential and Logarithmic Function 244
5.2.3 Floor and Ceiling Functions 245
5.2.4 Transcedental Function 247
5.2.5 Identity Function 247
5.2.6 Integer Value and Absolute Value Functions 247
5.2.7 Remainder Function 248
5.3 Composition of Functions 248
5.4 Inverse of a Function 250
5.5 Hashing Functions 256
5.6 Countable and Uncountable Sets 257
5.7 Characteristic Function of a Set 259
5.8 Permutation Function 261
5.9 Growth of Functions 262
5.10 The Relation Θ 262
Summary 267
Suggested Readings 269

6. LATTICE THEORY 271–304

6.1 Introduction 271
6.2 Partial Ordered Sets 272
6.2.1 Some Important Terms 273
6.2.2 Diagramatical Representation of a Poset (Hasse Diagram) 275
6.2.3 Isomorphism 276
6.2.4 Duality 278
6.2.5 Product of two Posets 280
6.3 Lattices as Posets 282
6.3.1 Some Properties of Lattices 283
6.3.2 Lattices as Algebraic Systems 284

6.3.3 Complete Lattice 290
6.3.4 Bounded Lattice 290
6.3.5 Sublattices 291
6.3.6 Ideals of Lattices 291
6.4 Modular and Distributive Lattices 292
Summary 302
Suggested Readings 304

7. BOOLEAN ALGEBRAS AND APPLICATIONS 305–354
7.1 Introduction 305
7.2 Boolean Algebra (Analytic Approach) 306
7.2.1 Sub-Boolean Algebra 308
7.2.2 Boolean Homomorphism 309
7.3 Boolean Functions 318
7.3.1 Equality of Boolean Expressions 319
7.3.2 Minterms and Maxterms 319
7.3.3 Functional Completeness 320
7.3.4 NAND and NOR 320
7.4 Combinatorial Circuits (Synthesis of Circuits) 326
7.4.1 Half-Adder and Full-Adder 326
7.4.2 Equivalent Combinatorial Circuits 328
7.5 Karnaugh Map 331
7.5.1 Don't Care Conditions 334
7.5.2 Minimization Process 335
7.6 Finite State Machines 344
Summary 347
Suggested Readings 352

8. FUZZY ALGEBRA 355–392
8.1 Introduction 355
8.2 Crisp Sets and Fuzzy Sets 357
8.3 Some Useful Definitions 360
8.4 Operations of Fuzzy Sets 362
8.5 Interval-Valued Fuzzy Sets (I-V Fuzzy Sets) 367
8.5.1 Union and Intersection of two I–V Fuzzy Sets 368

8.6 Fuzzy Relations 369
8.6 Fuzzy Measures 373
8.7.1 Belief and Plausibility Measures 373
8.7.2 Probability Measure 374
8.7.3 Uncertainty and Measures of Fuzziness 374
8.7.4 Uncertainty and Information 375
8.8 Applications of Fuzzy Algebras 376
8.8.1 Natural, Life and Social Sciences 376
8.8.2 Engineering 378
8.8.3 Medical Sciences 381
8.8.4 Management Sciences and Decision Making Process 382
8.8.5 Computer Science 383
8.9 Uniqueness of Uncertainty Measures 384
8.9.1 Shannon's Entropy 384
8.9.2 U-uncertainty 386
8.9.3 Uniqueness of the U-uncertainty for Two-Value Possibility Distributions 388
Summary 389
Suggested Readings 390

9. FORMAL LANGUAGES AND AUTOMATA THEORY 393–428
9.1 Introduction 393
9.2 Formal Languages 396
9.2.1 Equality of Words 397
9.2.2 Concatenation of Languages 398
9.2.3 Kleene Closure 399
9.3 Grammars 403
9.3.1 Phase-structure Grammar 406
9.3.2 Derivations of Grammar 406
9.3.3 Backus-Normal Form (BNF) or Backus Naur Form 407
9.3.4 Chomsky Grammar 410
9.3.5 Ambiguous Grammar 411

9.4 Finite-State Automation (FSA) 413
9.4.1 Counting to Five 414
9.4.2 Process of Getting up in the Morning (Alarm) 414
9.4.3 Traffic Light 415
9.4.4 Vending Machine 416
9.5 Finite-State Machine (FSM) 416
9.6 Finite-State Automata 418
9.6.1 Deterministic Finite-State Automata (DFSA) 418
9.6.2 Nondeterministic Finite-State Automata 419
9.6.3 Equivalent Nondeterministic Finite State Automata 420
Summary 424
Suggested Readings 428

10. THE BASICS OF GRAPH THEORY 429–480
10.1 Introduction 429
10.2 Graph. What is it? 430
10.2.1 Simple Graph 430
10.2.2 Graph 433
10.2.3 Loops 436
10.2.4 Degree of Vertices 436
10.2.5 Equivalence Relation 441
10.2.6 Random Graph Model 442
10.2.7 Isolated Vertex, Pendent Vertex and Null Graph 442
10.3 Digraphs 443
10.4 Path, Trail, Walk and Vertex Sequence 446
10.5 Subgraph 447
10.6 Circuit and Cycle 447
10.7 Cycles and Multiple Paths 449
10.8 Connected Graph 449
10.9 Spanning Subgraph and Induced Subgraph 450
10.10 Eulerian Graph (Eulerian Trail and Circuit) 450
10.11 Hamiltonian Graph 451
10.12 Biconnected Graph 452

10.13 Algebraic terms and operations used in Graph Theory 453
10.13.1 Graphs Isomorphism 453
10.13.2 Union of two Graphs 455
10.13.3 Intersection of two Graphs 455
10.13.4 Addition of two Graphs 456
10.13.5 Direct Sum or Ring Sum of two Graphs 456
10.13.6 Product of two Graphs 457
10.13.7 Composition of two Graphs 457
10.13.8 Complement of a Graph 457
10.13.9 Fusion of a Graph 458
10.13.10 Rank and Nullity 459
10.13.11 Adjacency Matrix 459
10.13.12 Some Important Theorems 460
10.14 Some Popular Problems in Graph Theory 465
10.14.1 Tournament Ranking Problem 465
10.14.2 The Königsberg Bridge Problem 467
10.14.3 Four Colour Problem 467
10.14.4 Three Utilities Problem 468
10.14.5 Traveling - Salesman Problem 468
10.14.6 MTNL'S Networking Problem 470
10.14.7 Electrical Network Problems 470
10.14.8 Satellite Channel Problem 471
10.15 Applications of Graphs 471
Summary 475
Suggested Readings 480

11. TREES **481–520**
11.1 Introduction 481
11.2 Definitions of a Tree 482
11.3 Forest 483
11.4 Rooted Graph 484
11.5 Parent, Child, Sibling and Leaf 485
11.6 Rooted Plane Tree 485
11.7 Binary Trees 492

11.8	Spanning Trees	494
11.9	Breadth – First Search and Depth – First Search (BFS and DFS)	496
11.10	Minimal Spanning Trees	504
	11.10.1 Kruskal's Algorithm (for Finding a Minimal Spanning Tree)	504
	11.10.2 Prim's Algorithm	509
11.11	Directed Trees	511
	Summary	516
	Suggested Readings	518

12. PLANAR GRAPHS **521–544**

12.1	Introduction	521
12.2	Geometrical Representation of Graphs	522
12.3	Bipertite Graph	524
12.4	Homeomorphic Graph	525
12.5	Kuratowski's Graphs	526
12.6	Dual Graphs	530
12.7	Euler's Formula	532
12.8	Outerplanar Graphs	535
	12.8.1 *k*-outerplanar Graphs	536
	Summary	542
	Suggested Readings	543

13. DIRECTED GRAPHS **545–574**

13.1	Introduction	545
13.2	Directed Paths	547
13.3	Tournament	549
13.4	Directed Cycles	550
13.5	Acyclic Graph	554
13.6	Di-Orientable Graph	555
13.7	Applications of Directed Graphs	558
	13.7.1 Job Sequencing Problem	558
	13.7.2 To Design an Efficient Computer Drum	560
	13.7.3 Ranking of the Participants in a Tournament	562

13.8 Network Flows 564
13.9 Improvable Flows 565
13.10 Max-Flow Min-Cut Theorem 567
13.11 *k*-flow 568
13.12 Tutte's Problem 569
Summary 571
Suggested Readings 574

14. MATCHING AND COVERING 575–608
14.1 Introduction 575
14.2 Matching and Covering in Bipertite Graphs 577
14.2.1 Covering 582
14.3 Perfect Matching 584
14.4 Factor-critical Graph 588
14.5 Complete Matching 590
14.6 Matrix Method to Find Matching of a Bipertite Graph 592
14.7 Path Covers 595
14.8 Applications 596
14.8.1 The Personnel Assignment Problem 596
14.8.2 The Optimal Assignment Problem 601
14.8.3 Covering to Switching Functions 602
Summary 604
Suggested Readings 607

15. COLOURING OF GRAPHS 609–640
15.1 Introduction 609
15.2 Vertex Colouring 612
15.3 Chromatic Polynomial 613
15.3.1 Bounds of the Chromatic Number 614
15.4 Exams Scheduling Problem 617
15.5 Edge Colouring 625
15.6 List Colouring 630
15.7 Greedy Colouring 631
15.8 Applications 635
15.8.1 The Time Table Problem 635

15.8.2 Scheduling of Jobs 636
15.8.3 Ramsey Theory 637
15.8.4 Storage Problem 637
Summary 638
Suggested Readings 639
References **641–642**
Index **643–648**

Check for updates

0

Preliminaries

Euclid (330–275 B.C.)

Euclid is known as the father of Geometry. A little is known about Euclid's life. He taught at the university of Alexandria and known as the founder of Alexandrian School of Mathematics. When the Egyption ruler King Ptolemy asked Euclid, if there were an easier way to learn geometry then by studying the elements, he replied, "There is no royal road to geometry." "No work, except for the Bible, has been more widely read, studied, or edited, according to J.E. Lughtner of Western maryland College, Westminister Maryland." More than 2000 editions of the work have appeared since the first printed one in 1482; however, no extant copy of the elements dates from Euclid's own time.

To start a detailed reading we must have some normal input. In this chapter we plan to discuss certain results which we feel the reader would already be familiar with. The idea is not to go into details but only to touch, upon the concepts to make the text self contained. The best place to start would, of course, be from the beginning. This chapter gives us normal input to start with.

0.1 Numbers

The ancient culture began mathematics by counting, keeping tallies. The early people need to keep records led to the use of tokens, small clay objects, to correspond to quantities of goods. Around 3100 B.C. ancient accountants began abstracting quantity from the objects being counted, and numbers were born. Once **natural numbers**, 1,2,3, ... were

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_0

available, manipulating them led to positive fractions known to be ancient Babylonians around 2000 B.C., who also knew some square roots and cube roots.

The classical Greek geometers (400 B.C.) studied positive quantities which could be obtained from natural numbers by the processes of arithmatic operations like addition, subtraction, multiplication, division and taking their square and cube roots.

The number 0 was not known, upto 300 B.C., negative number came in existance around 600 A.D. but became acceptable only after 1600 A.D. The complex numbers won acceptance between early 1500s and 1800. A precise understanding of seal numbers was reached only in 1870s.

Since 1800 mathematics has developed many new systems of objects which can be manipulated in the same way as these classical sets of numbers. As the classical numbers, once the domain of numbers is expanded, after 1800 A.D. the work is called modern mathematics.

$\mathbb{N}$ is the set of **natural** or **counting numbers**:

$$1, 2, 3, 4, 5, ...$$

$\mathbb{Z}$ is the set of integers

$$..., -3, -2, -1, 0, 1, 2, 3, ...$$

Obtained from $\mathbb{N}$ by including 0 (zero) and the negatives of the natural numbers.

$\mathbb{Q}$ is the set of **rational numbers**, that is, the set of all fractions (a/b) where a and b are integers and $b \neq 0$.

Note: (*i*) Rational (Ratio + nal)

(*ii*) $\mathbb{Z}$ as being a subset of Q by identifying an integer a with the fraction $a/1$. Q is large enough so that every non-zero integer has an inverse, or reciprocal in Q.

$\mathbb{R}$ is the set of **real numbers**. $\mathbb{R}$ is the set of all infinite decimals or as the set of coordinates on a line.

Any rational number may be a real number; the decimal expansion of a fraction a / b is obtained by the familiar process of dividing b into a.

$\mathbb{C}$ denotes the complex numbers. A complex number is an expression of the form $a + ib$ where a, b are real and $i = \sqrt{-1}$.

George Cantor, in the 1870s used the idea of numbers as a label associated with an **equivalence class** of sets. It is a basic mathematical strategy at work is the definition of rational numbers (Q), is a notion of dividing a set S up into equivalence classes, An equivalence class is a relation in a set S which satisfies:

(*i*) An element in S is equivalent to itself (reflexive property) *i.e.*, $\forall\ a$ in S, $a \approx a$;

(*ii*) If one element in S is equivalent to a second element is S then the second element is equivalent to the first (symmetric property) *i.e.*, $\forall\ a$ and b in S, if $a \approx a$ then $b \approx a$;

(*iii*) If one element in S is equivalent to a second element is S, and the second to third, then the first is equivalent to third (transitive property).

i.e., $\forall\ a$, b and c in S, if $a \approx b$ and $b \approx c$ then $a \approx c$.

When the equivalence relation is *one-to-one* correspondence, to study infinite numbers (infinite cardinal numbers). The cardinal numbers of any set A is also the cardinal number of all sets which are in one-to-one correspondence with set A.

The number No (aleph null), corresponds to the class of all sets which are in one-to-one correspondence with the set of all counting numbers, and the number c corresponds to the class of all sets which are in one-to-one correspondence with the set of all real numbers. A natural number in is **prime** if $n \geq 2$ and does not factor into the product of two natural numbers each smaller than n.

0.2 Euclid's Algorithm

If a divides b, a is a common divisor of b and it is manifestly also the greatest, for no number greater than a will divide a, then, the lesser of the number a, b being continuously subtracted from the greater, some number will be left which will divide the one before it. This number which is left is the **greatest common divisor** of b and a, and any common divisor of b and a divides the greatest common divisor of b and a.

Given natural numbers a and b, applying the division theorem successively, we get as follows:

$$\begin{aligned} b &= aq_1 + r_1, \\ a &= r_1q_2 + r_2, \\ r_1 &= r_2q_3 + r_3, \\ r_2 &= r_3q_4 + r_4, \\ &\vdots \quad \vdots \quad \vdots \\ r_{n-2} &= r_{n-1}q_n + r_n, \\ r_{n-1} &= r_nq_{n+1} + 0 \end{aligned}$$

If n is such that r_n divides r_{n-1}, then r_n is the greatest common divisor of a and b.

Illustration

For 78 and 32, we get

$$\begin{aligned} 78 - 32 &= 46, \\ 46 - 32 &= 14, \\ 32 - 14 &= 18, \\ 18 - 14 &= 4, \\ 14 - 4 &= 10, \\ 10 - 4 &= 6, \\ 6 - 4 &= 2, \end{aligned}$$

and 2 divides 4. So 2 is the greatest common divisor of 78 and 32.

0.3 Fundamental Theorem of Arithmetic

A natural number $p > 1$ is prime if the only divisor of p greater than 1 is p itself. We can note that 1 is not prime, by convention. Primes are the building blocks of natural numbers and can be illustrated by this theorem.

Theorem: *Any natural number > 1 factors into a product of primes.*

Proof: If $n > 1$ is prime, then it is a product of primes.

otherwise, $n = ab$,

with $1 < a < n$ and $1 < b < n$.

By induction

$a = p_1 \ldots p_r$, a product of primes

and also $b = q_1 \ldots q_s$, a product of primes

so $n = ab = p_1 \ldots p_r \cdot q_1 \ldots q_s$, a product of primes.

Fundamental Theorem of Arithmetic: Any natural number $n \geq 2$ factors uniquely into a product of primes.

Proof: The theorem can be proved by induction, suppose the result is true for all numbers $< a$. Let $a = p_1 \ldots p_n$ and also $a = q_1 \ldots q_n$ be factors of a into product of primes.

If $a = p_1$ is prime, then m and $n = 1$

and $p_1 = q_1$ (since the prime cannot factor into a product of two or more primes).

Hence theorem is true for a is not prime.

We further assume that a is not prime.

Let p_1, the leftmost prime in the first factorisation of a, is equal to some prime q_j the second factorization. The a / p_1 is a natural number ≥ 2 and $a / p_1 < a$.

Since $p_1 = q_j$, we get two factorizations of a / p_1, as

$$\frac{a}{p_1} = p_2 \ldots p_n$$

and

$$\frac{a}{p_1} = q_1 \ldots q_{j-1} \cdot q_{j+1} \ldots q_m$$

By induction, the two factors of a / p_1 are the same. That is, the set of primes $\{p_2, \ldots, p_n\}$ is the same as the set of primes $\{q_1, \ldots, q_{j-1}, q_{j+1}, \ldots, q_m\}$. Since $p_1 = q_j$, The set of primes $\{p_1, p_2, \ldots, p_n\}$ in the same as the set primes $\{q_1, \ldots, q_{j-1}, \ldots, q_j, q_{j+1}, \ldots, q_m\}$

The two factors of a are same, and the result is true for any number a.

We prove that if $p_1 \ldots p_n = q_1 \ldots q_m$ then $p_1 = q_j \; \forall \; j$.

Then *if p is prime and p divides ab, then p divides a or p divides b.*

Theorem: $\sqrt{2}$ *is irrational.*

Proof: Let it is not true

$$\sqrt{2} = a / b \; \forall \; ab \in \mathbb{N}$$

Multiplying both sides by b and squaring, we get

$$2a^2 = b^2$$

Let 2^e be the power of 2 appearing is the factorization of a and 2^f the power of 2 appearing in the factorization of b.

Then since $2a^2 = b^2$, we have $2e + 1 = 2f$.

But the left side of this equation is odd, and the right side even, which is impossible.

Hence $\sqrt{2}$ is irrational.

■

0.4 Euclid's Theorem

Statement: *There are infinitely many primes.*

Proof: We suppose that the set of primes is finite is number, say $p_1, p_2, ..., p_r$ are all prime numbers.

We consider the number $m = p_1, p_2 ... p_r + 1$.

It must have a prime factor q.

If q were one of the primes $p_1, p_2, ..., p_r$ then since q divides m, and q divides $p_1, p_2 ... p_r$, therefore q must divide their difference, which is 1. But this is impossible.

So, q cannot be one of the primes $p_1, p_2, ..., p_r$ and must be a new prime.

This contradiction that $p_1, p_2, ..., p_r$ were all the primes. So the number of primes cannot be finite.

0.5 Congruence Modulo *m*

Two integers a and b are congruent modulo m, writtern $a \equiv b \pmod{m}$

If m divides $a - b$, or equivalently, if $b = a$ + some multiple of m.

As a special case that a number a is congruent to 0 (mod m), writtern as $a \equiv 0 \pmod{m}$, iff m divides a.

But the value of the congruence notations is not providing an alternative for notation m/a but is providing a highly suggestive notation to use in place of $m/a - b$.

e.g., $1325 \equiv 2 \pmod{9}$

$182 \equiv 119 \pmod{9}$

$143 \equiv 0 \pmod{13}$

$143 \equiv 0 \pmod{11}$

$144 \equiv 1 \pmod{11}$

$145 \equiv 2 \pmod{13}$

$-3 \equiv 27 \pmod{6}$

$-3 \equiv -333 \pmod{6}$

Note: (*i*) In a congruence mod m, m is called the modulus (plural: "moduli").

(*ii*) Any two integers are congruent modulo 1, so the modules 1 is not of much interest. For this reason the modulus will normally be ≥ 2.

(*iii*) The set of integers to which the integer a is congruent modulo m is the set $\{a + mk/k$ any integer$\}$.

0.6 Chinese Remainder Theorem

The Chinese Remainder Theorem is a result from elementary number theory about the solution of a system of simultaneous congruences. It is also an application of software design. The chinese mathematician Sun-tri wrote about the theorem in about 100 A.D. In parallel processing this theorem has an interesting consequences. The theorem can be proved by the following lemma:

Lemma: Let m and n be positive integers such that g.*c.d.* $(m, n) = 1$. Then for $a, b \in \mathbb{Z}$ the system.

$$x \equiv a \pmod{m}$$

$$x \equiv b \pmod{n}$$

has a solution. If x_1 and x_2 are two solutions of the system,

then $$x_1 \equiv x_2 \pmod{mn}.$$

Proof: The equation $x \equiv a \pmod{m}$ has a solution since $a + km$ satisfies the equation for all $k \in \mathbb{Z}$.

We must have an integer k_1 such that

$$a + k_1 m \equiv b \pmod{n}$$

This is equivalent to

$$k_1 m \equiv (b - a) \pmod{n}$$

has a solution for k_1.

Since m and n are relatively prime.

$\exists$ integers s and t such that $ms + nt = 1$.

Consequently, $(b - a)\, ms = (b - a) - (b - a)\, nt$

or $$[(b - a)s]\, m \equiv (b - a) \pmod{n}$$

Let $$k_1 = (b - a)\, s$$

We have to show that any two solutions are congruent modulo m, to do so, we consider c_1 and c_2 as two solutions of the system.

i.e., $c_i \equiv a \pmod{m}$

$c_i = b \pmod{n}$

for $i = 1, 2$ then

$c_2 = c_1 \pmod{m}$

$c_2 = c_1 \pmod{n}$

Therefore, both m and n divide $c_1 - c_2$.

Consequently,

$$c_2 \equiv c_1 \pmod{mn}.$$

To prove the above we solve the system

$x \equiv 3 \pmod{4}$

$x \equiv 4 \pmod{5}$

We use Euclid's algorithm, we find integers s and t such that $4s + 5t = 1$.

Two such integers are $s = -1$ and $t = 1$

Consequently,

$$x = a + k_1 m = 3 + 4k_1 = 3 + 4[(5 - 4)\ 4] = 19.$$

Chinese Remainder Theorem: Let $n_1, n_2, ..., n_k$ be positive integers such that g.c.d. $(n_i, n_j) = 1$.

for $i \neq j$. Then for any integers $a_1, ..., a_k$, the system

$x \equiv a_1 \pmod{n_1}$

$x \equiv a_2 \pmod{n_2}$

$\vdots$

$x \equiv a_k \pmod{n_k}$

has a solutions. Further more, any two solutions of the system are congruent modulo $n_1, n_2 \ldots n_k$.

Proof: The theorem can be solved by using mathematical induction on the number of equation in the systems.

If there are $k = 2$ equations, then the theorem is true.

Let the result is true for a system of k equations or less and that we want to find a solution of

$x \equiv a_1 \pmod{n_1}$

$x \equiv a_2 \pmod{n_2}$

$\vdots$

$x \equiv a_{k+1} \pmod{n_{k+1}}$

We consider first k equations. There exists a solution that is unique modulo $n_1 \ldots n_k$, (say a) since $n_1 \ldots n_k$ and n_{k+1} are relatively primes.

The system

$$x \equiv a \pmod{n_1 \ldots n_k}$$
$$x \equiv a_{k+1} \pmod{n_{k+1}}$$

has a solution that is unique modulo $n_1 \ldots n_{k+1}$.

To illustrate the theorem, we solve the system

$$x \equiv 3 \pmod 4$$
$$x \equiv 4 \pmod 5$$
$$x \equiv 1 \pmod 9$$
$$x \equiv 5 \pmod 7$$

from previous illustration we have found that 19 is a solution of the first two congruences and any other solution of the system is congruent to 19 (mod 20).

Hence we may reduce the system to a system of three congruences, *viz*

$$x \equiv 19 \pmod{20}$$
$$x \equiv 1 \pmod 9$$
$$x \equiv 5 \pmod 7$$

solving the next two equations, we may reduce the system.

$$x \equiv 19 \pmod{180}$$
$$x \equiv 5 \pmod 7$$

solving the next system we find that 19 is a solution for the system that is unique up to modulo 1260.

■

0.7 Fermat's and Euler Theorems

The **Euler ϕ-function** is the map $\boldsymbol{\phi: N \to N}$ defined by $\phi(\boldsymbol{n}) = 1$ for $n = 1$ and for $n > 1$, $\boldsymbol{\phi(n)}$ is the number of positive integers m with $1 \leq m < n$ and *g.c.d.*$(m, n) = 1$.

We must know that order of $U(n)$, the group of units in $\mathbb{Z}_{\boldsymbol{n}}$, is $\phi(\boldsymbol{n})$.

e.g., $| \boldsymbol{U}(12) | = \boldsymbol{\phi}(12) = 4$ since the number that are relatively prime to 12 are 1, 5, 7 and 11.

For any prime p, $\boldsymbol{\phi}(p) = p - 1$. We state the following theorem.

Let $\boldsymbol{U(n)}$ be the group of units in $\mathbb{Z}_{\boldsymbol{n}}$. Then $| \boldsymbol{U(12)} | = \boldsymbol{\phi(n)}$.

Euler's Theorem: Let a and n be integers such that $n > 0$ and $gcd\,(a, n) = 1$. Then $a^{\phi(n)} \equiv 1 \pmod{n}$.

Proof: The order of $\boldsymbol{U(n)}$ is $\boldsymbol{\phi(n)}$.

Consequently, $a^{\phi(n)} = 1$.

for all $a \in \boldsymbol{U(n)}$; or $a^{\phi(n)} - 1$ is divisible by n.

Therefore, $a^{\phi(n)} \equiv 1 \pmod{n}$.

Note: If $n = p$ then $\boldsymbol{\phi(p)} = p - 1$.

Fermat's Little Theorem: Let p be any prime number and suppose that p does not divide a, then

$$a^{p-1} \equiv 1 \pmod{p}$$

Furthermore, for any integer b,

$$b^p \equiv b \pmod{p}.$$

0.8 Exponents and Logarithms

If x is a positive integer, b^x is the product of x copies of b, then $b^x = b \times b \times b \times ... \times b$ (x factors). In this expression b is called the base and x the exponent. It is easy to deduce such properties as

$$b^x b^y = b^{x+y}$$

$$(b^x)^y = b^{xy}$$

$$(ab)^x = a^x b^x$$

Negative exponents are handled by defining $b^{-x} = \dfrac{1}{b^x}$ and $b^0 = 1$ having b is non-zero.

The multiplication rule leads us to define $b^{1/x}$ to be the x – th root of b.

The process of taking powers can be inverted. The logarithm of x so base be or $\log_b x$ is defined to be the number y such that $b^y = x$. It is clear that $\log_b x$ is not defined if b is 1.

The properties of logarithms follow from the elementary properties of exponents, particularly:

(*i*) $\log_b (uv) = \log_b u + \log_b v$,

(*ii*) $\log_b (u^{-1}) = -\log_b u$,

(*iii*) $\log_b (x^y) = y \log_b x$,

(*iv*) $\log_b 1 = 0$ for any b.

Illustration

If we evaluate $125 \log_5 2$

$$\begin{aligned} 125 \log_5 2 &= (5^3) \log_5 2 \\ &= 5^3 \log_5 2 \\ &= 5 \log_5 2^3 \\ &= 5 \log_5 8 \\ &= 8. \end{aligned}$$

0.9 Sums

The sum of first ten positive integers can be written as

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10$$

Or more briefly $1 + 2 + ... + 10$. It should be clear that each number in the sum is obtained by adding 1 to the preceding number. A more precise notation is $\sum_{i=1}^{10} i$.

In the same way, we can write

$$\sum_{i=1}^{5} f(i) = f(1) + f(2) + f(3) + f(4) + f(5)$$

This notation means first evaluate the expression after the $\sum$ [*i.e.*, $f(i)$] when $i = 1$, then when $i = 2$, ..., then when $i = 5$, and add the result.

We take an example,

We want to evaluate $\sum_{i=1}^{4} i^2$ and $\sum_{i=3}^{6} i(i+1)$ to do so,

$$\begin{aligned} \sum_{i=1}^{4} i^2 &= 1^2 + 2^2 + 3^2 + 4^2 \\ &= 1 + 4 + 9 + 16 \\ &= 30 \end{aligned}$$

$$\begin{aligned} \sum_{i=3}^{6} i(i+1) &= 3 \cdot 4 + 4 \cdot 5 + 5 \cdot 6 + 6 \cdot 7 \\ &= 12 + 20 + 30 + 42 \\ &= 104. \end{aligned}$$

Some Properties of Sums: The following properties of sums are true:

(*i*) If c is any given number, than $\sum_{i=1}^{n} = nc$.

(*ii*) If c is any given number and (a_i) is any sequence of length n, than

$$\sum_{i=1}^{n} (ca_i) = c\left(\sum_{i=1}^{n} a_i\right)$$

(*iii*) If (a_i) and (b_i) are any two sequences, both of length n, than

$$\sum_{i=1}^{n} (a_i + b_i) = \left(\sum_{i=1}^{n} a_i\right) + \left(\sum_{i=1}^{n} b_i\right)$$

(*iv*) If (a_i) is any sequence of length n, $1 \leq j < n$, then

$$\sum_{i=1}^{n} a_i = \sum_{i=1}^{j} a_i \sum_{i=j+1}^{n} a_i$$

0.10 Mapping

Let A and B be two non-empty sets. A relation f from A to B is called a function or a mapping (map) from A to B if for every $a \in A$, $\exists$ a unique $b \in B$ *s.t.*, $(a, b) \in f$.

We write $f : A \to B$ and express $f(a) = b$.

A is called **domain** and B the **co-domain** of f. Also b is called *image* or consequent of a under f and a is called *pre-image* or antecedent of b under f. f is called a 1–1 (**one-one**) map if

$$f(x) = f(y) \quad \Rightarrow \quad x = y$$

or

$$x \neq y \quad \Rightarrow \quad f(x) \neq f(y)$$

f is called an onto map if each $b \in B$ has a pre-image in A. In fact the subset of all those members of B which have pre-images under f is called **range** of f.

The map $f : A \to A$, *s.t.*, $f(x) = x \ \forall \ x \in A$ is called identity map. It is clearly one-one onto.

If $f : A \to B$, $g : B \to C$ be two maps, then the mapping $gof : A \to C$ defined by

$$gof(x) = g\{f(x)\}$$

is called **composition of mappings**. One can extend this definition to more than two mappings.

Two mappings $f : A \to B$, $g : A \to B$ would be equal iff

$$f(x) = g(x) \quad \forall\, x \in A$$

Mapping can be illustrated by the following theorem.

Theorem: *A map $f : A \to B$ is one-one onto iff $\exists$ a map $g : A \to B$ such that gof and fog are identity maps on A and B respectively.*

Proof: Let $f : A \to B$ be 1–1 onto.

Define $g : B \to A$, *s.t.*, $g(b) = a \quad \Leftrightarrow \quad f(a) = b$

Let $b_1, b_2 \in B$ s.t., $b_1 = b_2$.

Since f is onto, $\exists\ a_1, a_2 \in A$ *s.t.*, $f(a_1) = b_1, f(a_2) = b_2$

Thus $b_1 = b_2 \quad \Rightarrow \quad f(a_1) = f(a_2) \quad \Rightarrow \quad a_1 = a_2$

i.e., g is well defined.

Also then $gof : A \to A$ and $(gof)\ a = g\{f(a)\} = g(b) = a$ for any $a \in A$.

Hence gof is identity map on A.

Similarly fog will the identity map on B.

Conversely, let $f : A \to B$ be such that $\exists$ some map $g : B \to A$ *s.t.*, fog and gof are identity maps on B and A.

Now $\quad f(a_1) = f(a_2)$

$\Rightarrow \quad g\{f(a_1)\} = g\{f(a_2)\}$

$\Rightarrow \quad (gof)\ a_1 = (gof)\ a_2$

$\Rightarrow \quad a_1 = a_2 \qquad$ (gof is identity map)

Thus f is 1–1

Again, for any $b \in B$, $fog(b) = b$

$\Rightarrow \quad f\{g(b)\} = b$

i.e., $g(b)$ is required pre-image of b under f

i.e., f is onto.

Definition: This mapping $g : B \to A$ is called inverse of $f : A \to B$ and is denoted by f^{-1}.

We thus gather a mapping f is invertible iff it is 1–1 onto.

EXERCISE

1. Prove that every composite number is divisible by some prime number.
2. Get 366 in base 2.

   Hint:
   $$366 = 2.183 + 0$$
   $$183 = 2.91 + 1$$
   $$91 = 2.45 + 1$$
   $$45 = 2.22 + 1$$
   $$22 = 2.11 + 0$$
   $$11 = 2.5 + 1$$
   $$5 = 2.2 + 1$$
   $$2 = 2.1 + 0$$
   $$1 = 2.0 + 1, \quad \text{so} \quad 366 = (101101110)_2.$$
3. Find *gcd* of 78 and 32. [**Ans.** 2]
4. Show that if $b = aq + r$, then $(b, a) = (a, r)$.
5. If a divides bc and $(a, b) = 1$, then prove that a divides c.
6. Prove that if the natural number a is not a square, then $\sqrt{a}$ is irrational.
7. Show that $(100)^{1/3}$ is irrational.
8. Suppose $ar + bs = r$ and $ax + by = r$. Show that there is some integer n so that $x = r + n[a, b]/a, y = s - n[a, b]/b$.
9. Show that if $n > 4$ is not prime, than $(n - 1)! \equiv 0 \pmod{n}$.
10. Show that $2^{560} = 1 \pmod{561}$.

Suggested Readings

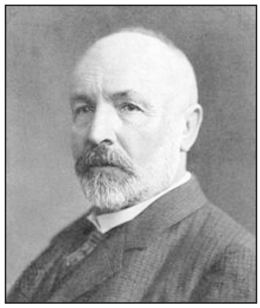
1. Gallian J., *Contemporary abstract Algebra,* 7th edition, Narosa, 2012
2. Telang, *Number theory,* TMH 2005
3. S.K. Yadav, *Discrete Mathematical Structures*, Satya Prakashan, N. Delhi.

❑❑❑

Check for updates

1

The Language of Sets



Georg Cantor
(1845–1918)

George Cantor was born in 1845 at St. Petersburg (Russia) in a successful merchant family. From early childhood he had a great interest in mathematics. The studied physics, mathematics and philosophy in the university of Halle. He was greatly influenced by Karl Weierstrass (1815–1897). He received his Ph.D. at 22 from the university of Berlin in number theory. In 1869, Cantor began his professional career as an unpaid lecturer at the university of Halle. Five years later, his revolutionary work on set theory was published. He developed on arithmetic of transfinite numbers analogous to that of finite numbers, creating a new area of study. He proved that the set of real numbers is uncountable and established the existence of infinitely many different transfinite cardinal numbers by ingenious method. He was a religious with deep interest in music, arts and philosophy too. He tried for a better paid position at the university of Berlin but blocked by Leopold Kroneckar (1823–1891), an eminent mathematician at the university, who always criticized Cantor's work Cantor died in a mental hospital in Halle in 1918.

1.1 Introduction

A set is a well defined collection of distinct objects, sets are of fundamental importance in mathematics. The very number systems that we studied in the preceding chapter were all discussed in terms of set of numbers. Many problems of discrete mathematics can conviently be expressed in terms of sets, especially finite sets. For this reason, we need to discuss the properties of sets and develop language to talk about them.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_1

The term 'set' is an undefined primitive of the mathematical system under study. The concept of a set is used in various disciplines and particularly often in mathematical computing. The set according to **Georg Cantor** (1895), a german mathematician (regarded as the father of set-theory), is a collection of definite well defined objects of perception of thought. A set may also be thought of as grouping together of single objects into a whole. The implications of the above descriptions are the widest possible. The language of sets is a means to study such collections is a well organised manner.

In fact, what is needed is that the objects should be distinct from each other and they should be distinguished from all those objects that do not form the set under consideration. The set may be a bunch of grapes, a tea set or it may consist of geometrical points or straight lines or may be a collection of sets too.

A class, a family, a collection and an aggregate may be some of the other terms, which more or less have the same meaning as a set.

In a set well defined objects are called elements (or members) of the set. There should be no ambiguities in determining whether or not a given object belongs to the set. For example, the vowels of the English alphabet form a (well defined) set, whereas beautiful cities in India do not form a set since its membership would be debatable.

The concept of a function and relation is extremely important in discrete mathematics. A function assigns to each element of a set exactly one element of a set. In this chapter, we will study the fundamental discrete structure on which all other discrete structures are built, namely, the set. Sets are used to group objects together.

1.2 Elements and Notations of Sets

The objects which constitute a particular set are said to be elements of the set or the members of the set. The particular relationship of an object to a set of which it is an element is called a relation or belonging.

A set is usually denoted by a capital italic letter, such as A, B, T, X *etc.* The elements of the set are represented by small italic letters such as a, b, c, x, y *etc.* Certain special sets of number system are frequently used and represented by particular symbols.

(*i*) **Set of natural numbers:** The positive integers, the counting numbers or the natural numbers are the positive whole numbers *viz.*, 1, 2, 3, 4 ..., The set of all these numbers is denoted by $\mathbb{Z}$.

(*ii*) **Set of non-negative numbers:** These are the numbers 0, 1, 2, ...

(*iii*) **Set of integers:** The integers are the numbers 0, ± 1, ± 2, ... The set of all integers is denoted by I or $\mathbb{Z}$.

(*iv*) **Set of even integers:** Any integer of the form 2^n, where n is an integer is termed as even integer. Thus, ± 2, ± 4 ... are even integers.

(*v*) **Set of odd integers:** Any integer of the form 2^{n-1} or 2^{n+1} where n is an integer is called an odd integer. Thus ± 1, ± 3, ± 5 ... are all odd integers.

(*vi*) **Set of rational numbers:** Any number of the form p / q where p and q are integers and $q \neq 0$ is called rational number. The set of rational numbers is denoted by $\mathbb{Q}$.

(*vii*) **Set of real numbers:** The set of which, the elements are all rational and irrational numbers is called the set of real numbers and denoted by $\mathbb{R}$.

Note: A real number which is not rational is called irrational. It is clear that $\sqrt{2}$ is irrational as it cannot be put in the form of p / q where p and q are integers and $q \neq 0$.

(*viii*) **Set of complex numbers:** Any number of the form $a + ib$ where $i = \sqrt{-1}$ and a, being real, is called complex number. The set of complex numbers is denoted by $\boldsymbol{C}$.

(*ix*) **Set of prime numbers:** A positive integer n other than 1, which is not divisible by an integer except 1 and n itself is called a prime number. Set of prime numbers is denoted by $\boldsymbol{P}$.

(*x*) **Set of composite numbers:** A positive integer which is not prime is called a composite number (integer).

1.3 Construction of Sets

There are two methods of construction of sets *viz.*:

(*a*) **Enumeration or Roster Form:** This method constitutes a list of objects forming the set, by writing the elements one after another between a pair of curly braces. During this process, a

list of objects can be completely exhausted or it may not be completely exhausted. Thus a set A with elements 1, 3, 5, 7, ... will be written as $A = \{1, 3, 5, 7, ...\}$.

(*b*) **Description or set builder form:** This method constitutes one or more properties that are satisfied by only those objects constitute the set we want to construct. This method points to two important features of sets:

(*i*) The order is which the objects are taken is immaterial.

(*ii*) Certain elements may possess some properties that are not possessed by others.

Further notations:

(*i*) The symbol $\in$ stands for "*belongs to*" or "*is an element of*".

(*ii*) The symbol $\notin$ stands for "*does not belong to*" or "*is not an element of*".

Let us illustrate the set,

We consider a set A and p, q certain objects. If p is an element of A, this indicates as $p \in A$. If p and q both belong to A then p, $q \in A$. instead of $p \in A$ and $q \in A$.

If q does not belong to A, then we write $q \notin A$.

(*iii*) The colon (:) or (|) will stand for "*such that*" thus, if in description method we describe a set B "consisting of all integers divisible by 3" then symbolically we write

$$B = \{x : x \text{ is an integer divisible by } 3\}$$

in the above notation x is merely a typical element without making out precisely which element it is. If P stands for the property which every element of set A has to satisfy, then set A is represented as

$$A = \{x : x \text{ satisfy } P\} \text{ or } A = \{x : P(x)\}$$

Where $P(x)$ means "*property P satisfied by every x of the set*" Thus, the set B above can be devoted by

$$B = \{x : x \in T, x \text{ is divisible by } 3\}$$

(*iv*) If the elements of any set B are referred to as any element x without marking out precisely which element it is, then x is called **variable** of set B. Thus variable is a symbol which can take the place of any element in the given set.

(*v*) The set B of which x is the variable is called the **range** of variable.

1.4 Types of Sets

The common types of sets are:

(*i*) **Singleton set:** A set consists of only one element is called a singleton set. The sets given by {0}, {1}, {ϕ}, {*a*} are all singleton sets.

Note: If an element (object) appears is a set more than once, we shall not take it to mean more than one element *i.e.*, the set {0, 0, 1, 1) is the same as {0, 1}, similarly the set {1, 1, 1} simply means {1}, a singleton set.

(*ii*) **Finite set:** A set consists of finite number of elements is called a finite set.

If $A = \{5, 7, 9, 11\}$ and $B = \{4, 8, 16, 32, 64\}$

A and B both contain a finite number of elements *i.e.*, 4 objects in A and 6 in B. Hence A and B are finite sets.

(*iii*) **Infinite set:** If the number of elements in a set is infinite, the set so, is called an infinite set. *e.g.*, The set of all natural numbers given by $\mathbb{N} = \{1, 2, 3, ...\}$ is an infinite set.

Similarly the set of all rational numbers between 0 and 1, given by $A = \{x : x \in Q,\ 0 < x < 1\}$ is infinite set.

(*iv*) **Null set:** A set consists of no element, is called an empty set or Null set or void set. It is represented by ϕ. *e.g.*, $\phi = \{x : x \neq x\}$.

Note: {ϕ} is not a Null set. It is singleton set.

(*v*) **Equal sets:** Two sets A and B are said to be equal sets if they consist of same elements *i.e.* If each element of set A is an element of B and each element of B is an element of A than $A = B$.

(*vi*) **Subsets of a given set:** If A be a given set. Any set B, each of which element is also an element of A, is said to be contained in A and is called a subset of A.

The symbol $\subseteq$ stands for "*is contained in*" or "*is subset of*". Thus, if "B is contained in A" or "we write $B \subseteq A$.

When B is subset of A, we can say "A contains B" or "A is superset of B".

The symbol $\supseteq$ is read for "contains".

$A \supseteq B$ means "A contains B".

(*vii*) **Proper subset:** If B is a subset of A and $B \neq A$, then B is called proper subset of A *i.e.*, if each element of B is an element of A and there is at least one element of A which is not an element of B, then B is called a proper subset of A. "*is proper subset of*" is symbolically represented by $\subset$. Also the empty set ϕ is a proper subset of every set except itself.

Note: (*i*) The symbol $\in$ represents "*the relation of belonging of element to a set*".

(*ii*) The symbol $\subset$ or $\subseteq$ represent "*the relation of a set to another set*".

(*viii*) **Power set:** The set of all subsets of a given set A, is called the power set of A. Power set of A is denoted by $P(A)$.

If the set $A = \{a, b, c\}$ then its subsets are ϕ, $\{a\}$, $\{b\}$, $\{c\}$, $\{a, b\}$, $\{b, c\}$, $\{c, a\}$, $\{a, b, c\}$.

$\therefore \quad P(A) = \phi, \{a\}, \{b\}, \{c\}, \{a, b\}, \{b, c\}, \{c, a\}, \{a, b, c\}$

Note: Number of subsets of a set be 2^n (where n is the number of elements of that set).

(*ix*) **Cardinality of a set:** The cardinality of a set A is the number of elements in A and is denoted by $| A |$. *e.g.*, Considering the following:

(*i*) If $A = \{x, y, z\}$ then $| A | = 3$

(*ii*) If $A = \{x : x$ lies between 0 to 30 and divisible by 4$\}$ then $| A | = 7$.

(*x*) **Redundant description of a set:** In a set, if some elements are repeated, it is only one of them which is understood as belonging to the set. If $A = \{1, 2, 3,\}$, $B = \{2, 3, 1, 2, 3\}$ then $| A | = | B | = 3$.

(*xi*) **Universal set:** A set which is super set of all the sets under consideration is called a universal set and devoted by Ω or $\mathbb{S}$ or $\mathbb{U}$.

Note: Universal set can be chosen arbitrarily for discussion but once chosen, it is fixed for the discussion.

e.g., Let $A = \{1, 2, 3\}$

$B = \{3, 4, 6, 9\}$

$C = \{0, 1\}$

Then we can take

$U = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ as universal set.

(*xii*) **Cortesian Products:** Since the sets are unordered, a different structure is needed to represent ordered collections which is provided by ordered *n*-tuples. The ordered *n*-tuple $(a_1, a_2, a_3, ..., a_n)$ is defined as the ordered collection that has a_1 as its first element, a_2 as its second element, ..., and an as *n*th element.

Let A and B be two sets. The Cartesian product of A and B, denoted by $A \times B$, is the set of all ordered pairs (a, b), where $a \in A$ and $b \in B$. Hence $A \times B = \{(a, b) : a \in A \wedge b \in B\}$.

The Cartesian product of the sets $A_1, A_2, A_3, ..., A_n$, denoted by $A_1 \times A_2 \times ... \times A_n$, is the set of ordered *n*-tuples $(a_1, a_2, ..., a_n)$, where $a_i \in A_i \; \forall \; i = 1, 2, 3, ..., n$

$A_1 \times A_2 \times A_3 \times ... \times A_n = \{(a_1, a_2, a_3, ..., a_n) : a_i \in A_i \; \forall \; i = 1, 2, ... n\}$.

An Important Theorem: If A and B are two sets such that $A \subseteq B$ and $B \subseteq A$, then $A = B$.

Proof: Let $A \subseteq B$ and $x \in A$

$\therefore \quad x \in A \quad \Rightarrow \quad x \in B$ ...(*i*)

But $B \subseteq$ x

$\therefore \quad x \in B \quad \Rightarrow \quad x \in A$ ...(*ii*)

By (*i*) $x \in A \quad \Rightarrow \quad x \in B$ and by (*ii*) $x \in B \quad \Rightarrow \quad x \in A$.

$\therefore \quad x \in A \quad \Leftrightarrow \quad x \in B \quad \Rightarrow \quad A = B$.

Note: This theorem is used to prove the equality of two sets.

Example 1

Test the truth and validity of the following statements:

(*i*) $0 \in \{0, 1\}$;

(*ii*) $\{2\} \in \{1, 2, 3\}$;

(*iii*) $p \subset \{p, q(p)\}$;

(*iv*) $\{\phi\} \subset \{p, q\}$.

Solution

(*i*) The statement is true because 0 is an element of $\{0, 1\}$.

(*ii*) The statement is false because $\{2\}$ is not an element of $\{1, 2, 3\}$.

(*iii*) The statement is false because p is not a set but simply an element of the given set $\{p, q(p)\}$.

(*iv*) This statement is false, for the set $\{\phi\}$ consists of an element ϕ which is not an element of the given set $\{p, q\}$. ■

Example 2

Which of the following are finite or infinite sets:

(*i*) $A = \{2, 2, 2, 2, \ldots \textit{upto infinite times}\}$.

(*ii*) $B = \{\boldsymbol{x} : x \in I \textit{ and } x \textit{ is odd}\}$.

Solution

(*i*) In a set, if an element is repeated any times, it is to be counted once only. The set $A = \{2\}$, which is singleton. Hence it is a finite set.

(*ii*) Since all the elements of B are all odd integers, and there are infinite number of distinct odd integers. Hence B is an infinite set. ■

Example 3

Which of the following sets are Null sets and singleton sets.

(*i*) $A = \{x : x \in \mathbf{N}, 3x^2 - 14x - 5 = 0\}$;

(*ii*) $B = \{x : x^3 = 8, 2x = 5\}$;

(*iii*) $C = \{x : x \in \mathbf{N}, x + 5 = 5\}$.

Solution

(*i*) $3x^2 - 14x - 5 = 0$ gives $x = 5, -1/3$

But $x \in \mathbf{N}$

then only $x \in A$ is 5

Hence $A = \{5\}$ which is a **singleton set**.

(*ii*) The given equation $2x = 5 \Rightarrow x = 5/2$

But $x = 5/2$ does not satisfy $x^3 = 8$

There is no x which satisfies both equation

Hence B is a **null set**.

(*iii*) The equation $x + 5 = 5 \Rightarrow \boldsymbol{x} = 0$

$0 \notin N$

Hence there is no element x belonging to C.

Hence C is a **null set.** ■

Example 4

If $B \supset A, A \supset C, C \supset B$ *then prove that* $A = B = C$.

Solution

Let $x \in B, B \supset C$

$\therefore \quad x \in B \quad \Rightarrow \quad x \in C.$

$\Rightarrow \quad x \in A$ ...(*i*)

$\therefore \quad A \supset C$

But $B \supset \text{A}$

$\therefore \quad x \in A \quad \Rightarrow \quad x \in B$ ...(*ii*)

from (*i*) and (*ii*), we get

$x \in A \quad \Leftrightarrow \quad x \in B$

Again, let $x \in C$, But $A \supset C$

$\therefore \quad x \in C \quad \Rightarrow \quad x \in A$

$\Rightarrow \quad x \in B$

$\therefore \quad B \supset A$

$\therefore \quad C \subseteq B$ ...(*iii*)

But $B \subset C$

Hence from eqn. (*iii*) $B = C$

Thus $A = B = C$ ■

Example 5

A set consisting of n distinct elements prove that it yields $(2^n - 1)$ *proper subsets.*

Solution

There would be $(n + 1)$ types of subsets *i.e.*, subsets consisting of no element *i.e.*, ϕ, only one element, only two elements ..., only $n - 1$ elements and all the n elements.

Number of sets having no element = ${}^nC_0$

Number of sets having one element only = ${}^nC_1$

Number of sets having two eleemnt only = ${}^nC_2$

... ...

... ...

Number of sets having r elements = ${}^nC_r$
Number of sets having n elemens = ${}^nC_n$
Total number of subsets = ${}^nC_0 + {}^nC_1 + {}^nC_2 + ... + {}^nC_r + ... + {}^nC_n = 2^n$.
The set consisting of all the elements is not a proper subset.
Hence the number of proper subsets = $2^n - 1$. ■

Example 6

Determine the power sets of the following sets:

(*i*) $\{a\}$ (*ii*) $\{\{a\}\}$ (*iii*) $\{\phi, \{\phi\}\}$.

Solution

(*i*) We consider the set $A = \{a\}$
Here $n = 1$
The power set has $2^1 = 2$ elements
$\therefore \quad P(A) = P\{a\} = \{\phi, \{a\}\}$.

(*ii*) The set $B = \{a\}$ has only one element *i.e.*, $\{a\}$
$\therefore$ $n = 1$ and power set has $2^1 = 2$ elements
$\therefore \quad P(B) = P(\{\{a\}\}) = \{\phi, \{a\}\}$

(*iii*) The set $C = \{\phi, (a)\}$ has 2 elements *i.e.*, $n = 2$
$\therefore$ The power set has $2^2 = 4$ elements
$P(C) = (\{\phi, \{\{\phi\}\}) = \{\phi, \{\phi\}, \{\phi\}\}, \{\phi, \{\phi\}\}$. ■

Example 7

What is the Cartesian product of

(*i*) $A = \{1, 2\}, \quad B = \{a, b, c\}$

(*ii*) $A = \{0, 1\}, \quad B = \{1, 2\}$ *and* $C = \{0, 1, 2\}$.

Solution

(*i*) $A = \{1, 2\}, \quad B = \{a, b, c\}$
$A \times B = \{(1, a), (1, b), (1, c), (2, a), (2, b), (2, c)\}$

(*ii*) The Cartesian product $A \times B \times C$ consists of all ordered triples (a, b, c) where $a \in A$, $b \in B$ and $c \in C$

Hence $A \times B \times C = \{(0, 1, 0), (0, 1, 1), (0, 1, 2), (0, 2, 0), (0, 2, 1), (0, 2, 2), (1, 1, 0), (1, 1, 1), (1, 1, 2), (1, 2, 0), (1, 2, 1), (1, 2, 2)\}$. ■

1.5 Set Operations

This section includes certain operations of sets, yielding a new set in existence.

1.5.1 Intersection of Sets

The intersection of two sets A and B can be written as $A \cap B$. It is the set consisting of all the elements which belong to both the sets A and B.

As a symbol, $A \cap B = \{x : x \in A \text{ and } x \in B\}$ from definition above, the following exist for A and B.

(i) $A \cap B = B \cap A$ (commutativity)

(ii) $A \cap A = A$

(iii) $A \cap \phi = \phi$

For the first equality,

for any x, *s.t.*, $x \in A \cap B \quad \Leftrightarrow \quad x \in \{x : (x \in A) \text{ and } (x \in B)\}$

$\Leftrightarrow \quad (x \in A) \cap (x \in B)$

$\Leftrightarrow \quad (x \in B) \cap (x \in A)$

$\Leftrightarrow \quad \{x : (x \in B) \quad \text{and} \quad (x \in A)\}$

$\Leftrightarrow \quad x \in B \cap A$

Since $a \cap B$ is a set, than we can consider its intersection with another set C (say)

so that $(A \cap B) \cap C = \{x : x \in A \cap B \text{ and } x \in C\}$

using $(x \in A \text{ and } x \in B) \text{ and } x \in C$

$\Leftrightarrow \quad x \in A \text{ and } (x \in B \text{ and } x \in C)$

We can easily show that,

(iv) $(A \cap B) \cap C = A \cap (B \cap C)$[Associativity]

Indexing above we may write;

for an indexed set $A = \{A_1, A_2, A_3, \ldots A_n\} = \{A_i\}_i \in i_n$

Where $\quad \text{In} = \{1, 2, 3, \ldots, n\}$

We may write it as $A_1 \cap A_2 \cap A_3 \cap ... \cap A_n = \bigcap_{i=1}^{n} A_i = \bigcap_{i \in I_n} A_i$

Generally, for any index set J

$$\bigcap_{i \in j} A_i = \{x : x \in A_i \ \forall \ i \in j\}$$

Note: The intersection operation ($\cap$) is also termed as '**meet**' or '**logical product**' of sets A and B.

1.5.2 Union of Sets

The union of two sets A and B is written as $A \cup B$. It is the set of all elements which are the members of the set A or the set B or both.

Symbolically,

$$A \cup B = \{x : x \in A \quad \text{or} \quad x \in B\}$$

From definition

(*i*) $A \cup B = B \cup A$ (Commutativity)

(*ii*) $A \cup \phi = A$

(*iii*) $A \cup A = A$

(*iv*) $(A \cup B) \cup C = A \cup (B \cup C)$ (Associativity)

Associativity shows that

$$(A \cup B) \cup C = A \cup B \cup C$$

For above equalities, we have

for any x, $x \in A \cup A \Leftrightarrow x \in \{x : x \in A \quad \text{or} \quad x \in A\}$

$\Leftrightarrow \quad x \in A \quad \text{or} \quad x \in A$

$\Leftrightarrow \quad x \in A$

$\Leftrightarrow \quad x \in \{x : x \in A\}$

$\Leftrightarrow \quad x \in A$

hence for any indexed set $A = \{A_i\}\ i \in j$,

then $$\bigcup_{i=j} A_i = \{x : x \in A_i \text{ for at least one } i \in j\}$$

for $j = I_n = \{1, 2, 3, ..., n\}$ we may write

$$\bigcup_{i=1}^{n} A_i = A_1 \cup A_2 \cup A_3, ..., \cup A_n$$

1.5.3 Disjoint Set (Mutually Exclusive)

Let A and B be two sets, then A and B are called disjoint if and only if they have no element in common. *i.e.*, $A \cap B = \phi$.

Let A be an indexed set $A = \{A_i\}_{i \in j}$

The set A is a disjoint collection if and only if,

$$A_i \cap A_j = \phi \quad \forall\, i, j \in j, i \neq j$$

1.5.4 Ordinary Difference of Sets ($A - B$)

Let A and B be two sets of all those elements of A which do not belong to B, is represented by $A - B$.

Thus $A - B = \{x : x \in A \text{ and } x \notin B\}$

for example if $A = \{a, b, c, d, e\}$, $B = \{d, e, l, m, n\}$

only elements of A which do not belong to B are a, b, c.

Hence $A - B = \{a, b, c\}$.

1.5.5 Complementation of Sets

If B is a subset of A, then the set of all those elements of A which do not belong to B is called the complement of B in A.

For example,

If $B = \{p, q, r\}$ and $A = \{p, q, r, s, t\}$ then the complement of B is A is $\{s, t\}$.

1.5.6 Universal Set and its Complement

If all the sets under study are subset of a particular set, that particular set is known as universal set. Infact every set may be regarded as a subset of universal set denoted by U explicitly given or not.

Let A be a given set and U be the universal set so that $A \subset U$. Set of all those elements of U that are not the elements of A is another proper subset of U and is called the complement of A in U.

Complement of A in U is denoted by A' or A^C.

If $A = U$ then $A' = \phi$

$\Rightarrow$ $(A')' = A$.

1.5.7 Symmetric Difference (Boolean Sum)

If we have two sets A and B. The symmetric difference of A and B is the set $A \oplus B$ defined by

$$A \oplus B = (A - B) \cup (B - A) \text{ or } x \in A \oplus B \Leftrightarrow x \in \{x : x \in A \ \overline{v} \ x \in B\}$$

Where $\overline{v}$ is called exclusive disjunction.

The following are important equalities of symmetric difference

(*i*) $A \oplus B = B + A$

(*ii*) $(A \oplus B) + C = A \oplus (B \oplus C)$

(*iii*) $A \oplus \phi = A$

(*iv*) $A \oplus A = \phi$

(*v*) $A \oplus A = (A \cap \sim B) \cup (B \cap \sim A)$

[$\sim$ is absolute complement and $\sim A = U - A$]

1.6 Venn Diagrams

A relationship between sets can be displayed by using Venn diagrams, named after the English logician **John Venn** (1834–1932). Venn gave diagrammatic representations to the set operations.

Frankly speaking, a Venn diagram is a schematic representation of a set by a set of points. The universal set U is represented by a set of points in a rectangle (or any other figure), and a subset (say A) of U is represented by the interior of a circle or some other simple closed curve inside the rectangle (As shown in Fig. 1.1). The shaded area represent the sets indicated below each figure.

The Venn diagram for $A \subseteq B$ and $A \cap B = \phi$ are also given.

With the help of Venn diagram it is easy to see that the following cases hold easily;

$$A \cup B = B \cup A, A \cap B = B \cap A \quad \text{and} \quad \sim(\sim A) = A$$

If, $A \subseteq B$, then

$$A - B = \phi, \quad A \cap B = A \quad \text{and} \quad A \cup B = B.$$

Note: Venn diagrams do not provide such relations that are true in general for all subsets of U.

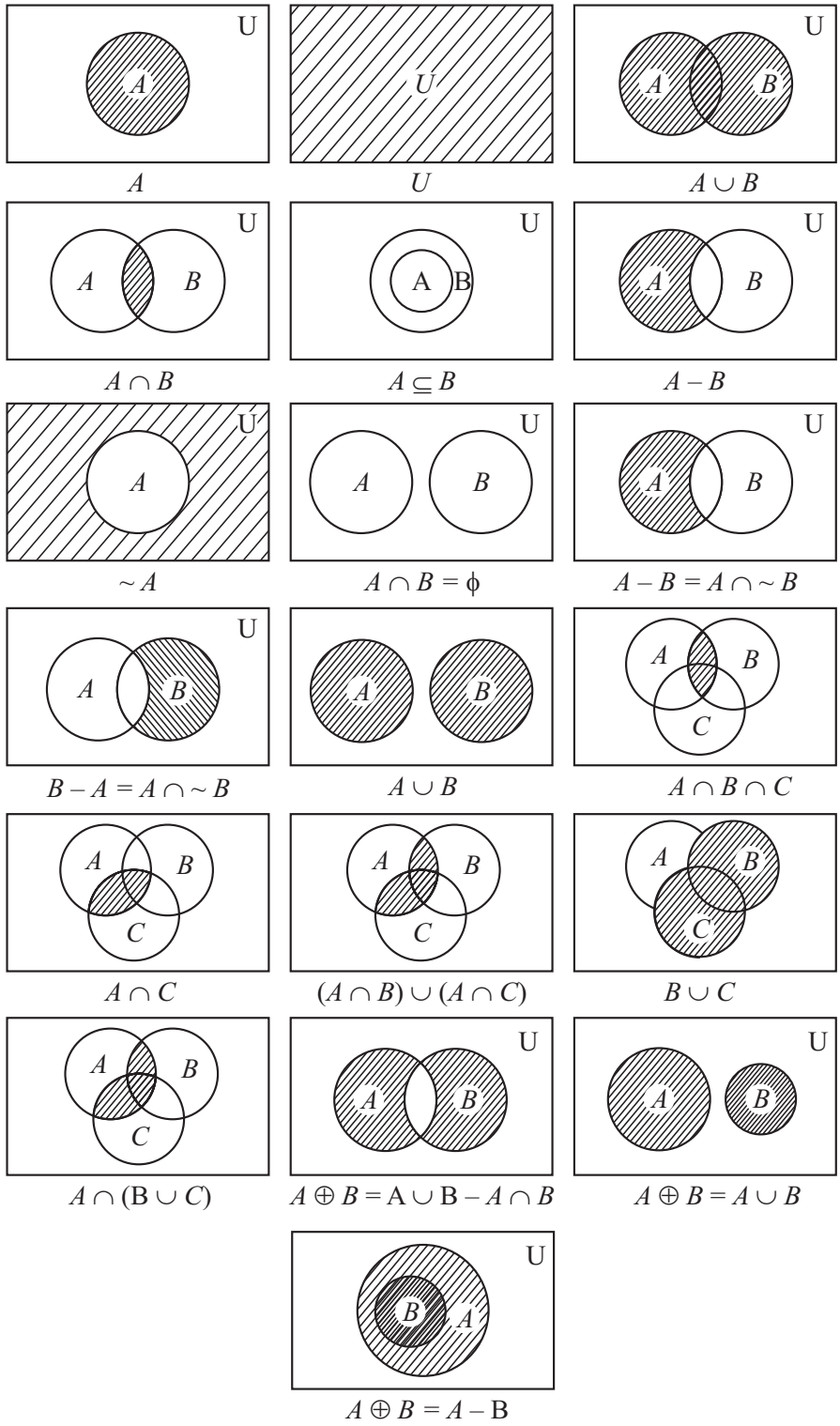


Fig. 1.1: Venn diagrams

Let us consider the Venn diagram given in above figure from first two diagrams it appears that

$$A \cup B = (A \cap \sim B) \cup (B \cap \sim A) \cup (A \cap B) \qquad ...(A)$$

From third Venn diagram it appears that

$$A \cup B = (A \cap \sim B) \cup (B \cap \sim A)$$

This equality is not true in general, although it happens to be true for the two disjoint sets A and B.

A formal proof of Eqn. (A) can be proved as,

For any x,

$$x \in A \cup B \quad \Leftrightarrow \quad x \in \{x : x \in A \text{ or } x \in B\}$$

$$x \in (A \cap \sim B) \cup (B \cap \sim A) \; \forall \; (A \notin B)$$

$\Leftrightarrow \quad x \in \{x : x \in (A \cap \sim B) \text{ or } x \in (B \cap \sim A) \text{ or } x \in (A \cap B)\}$

$\Leftrightarrow \quad x \in (A \cap \sim B) \text{ or } x \in (B \cap \sim A) \text{ or } x \in (A \cap B)$

$\Leftrightarrow \quad (x \in A \quad \text{and} \quad x \in \sim B) \text{ or } (x \in B \quad \text{and} \quad x \in \sim A)$

or $\qquad (x \in A \quad \text{and} \quad x \in B)$

$\Leftrightarrow \quad \{x \in A \quad \text{and} \quad (x \in \sim B \text{ or } x \in B)\} \text{ or } (x \in B \quad \text{and} \quad x \in \sim A)$

$\Leftrightarrow \quad (x \in A \quad \text{or} \quad x \in B) \quad \text{and} \quad (x \in A \text{ or } x \in \sim A)$

$\Leftrightarrow \quad (x \in A \text{ or } x \in B)$

$\Leftrightarrow \quad x \in \{x : x \in A \text{ or } x \in B\}$

$\Leftrightarrow \quad x \in A \cup B.$

From the third and fifth Venn diagrams it appears that

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C) \qquad ...(B)$$

Similarly, one can show that for any three sets A, B and C

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C) \qquad ...(C)$$

The equations (B) and (C) are also called the distributive laws of union and intersection.

For Eqn.(C) above,

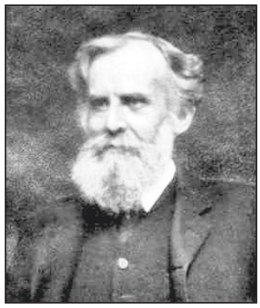
For any x,

$$x \in A \cup (B \cap C) \quad \Leftrightarrow \quad x \in \{x : x \in A \text{ or } x \in (B \cap C)\}$$

$\Leftrightarrow \quad x \in \{x : x \in A \text{ or } (x \in B \quad \text{and} \quad x \in C)\}$

$\Leftrightarrow \quad x \in \{x : x \in A \text{ or } x \in B \quad \text{and} \quad (x \in A \text{ or } x \in C)\}$

$\Leftrightarrow \quad x \in (A \cup B) \cap (A \cup C)$



John Venn (1834–1923)

John Venn (1834–1923) was born in a philanthropic family is Hull, England. After schooling at Highgate and Islington, in 1853, he entered Gonville and Caius College Cambridge, and graduated in Mathematics three years later. He was elected as the fellow of Royal society and a position he hold till death. In 1859 he was ordained in the Church of England, but after a brief period of Church work, he returned to Cambridge as a lecturer on moral sciences. In 1883 he gave up his priesthood and received a D.Sc. from Cambridge.

Venn was greately influenced by Boole's work on symbolic logic, Venn's masterpiece, symbolic logic (1881), clarifies the in-consistencies and ambiguities in Boole's ideas and notations. He employed geometric diagrams to represent logical arguments, a technique originated by Leibnitz and developed further by Euler. Venn added a rectangle to represent the universe of discourse. Venn published two important books. The Logic of Chance (1866) and The Principles of Empirical Logic (1889).

Example 8

Let $A = \{a, b, c, d, g\}$, $B = \{b, c, d, e, f\}$ and $C = \{b, c, e, g, h\}$ find $A \cup (B \cap C)$ and $(A \cup B) \cap (A \cup C)$.

Solution

$$B \cap C = \{b, c, e\}$$

$$\Rightarrow \quad A \cup (B \cap C) = \{a, b, c, d, e, g\}$$

$$A \cup B = \{a, b, c, d, e, f, g\}$$

$$A \cup C = \{a, b, c, d, e, g, h\}$$

$$(A \cup B) \cap (A \cup C) = \{a, b, c, d, e, g\}$$

$$= A \cup (B \cap C)$$

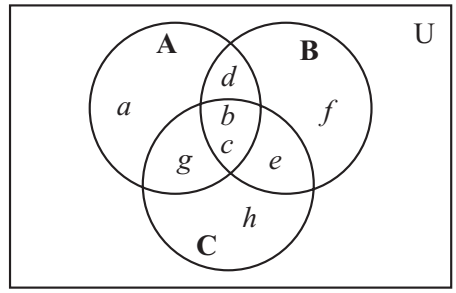


Fig. 1.2

Example 9

Let $U = \{a, b, c, ..., z\}$. *Find the complement of the set* $A = \{a, e, i, o, u\}$ *and* $B = \{a, c, d, e, ..., w\}$.

Solution

$A' = U - A$ = set of all consonants in the English alphabet and $B' = U - B = \{b, x, y, z\}$. ■

Example 10

Let $A = \{a, b, x, y, z\}$, $B = \{c, d, e, x, y, z\}$ *and* $U = \{a, b, c, d, e, w, x, y, z\}$. *Find* $(A \cup B)'$ *and* $A' \cap B'$.

Solution

$$A \cup B = \{a, b, c, d, e, x, y, z\}$$
$$(A \cup B)' = \{w\}$$
$$A' = \{c, d, e, w\}$$
$$B' = \{a, b, w\}$$
$$A' \cap B' = \{w\} = (A \cup B)'$$

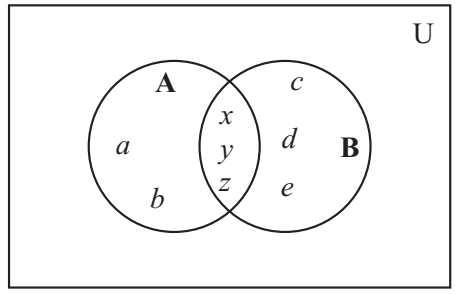


Fig. 1.3

1.7 Some Basic Results

The following results are basic results of set theory.

(*i*) $A \cup A = A$ (*ii*) $A \cap A = A$

(*iii*) $A \cup B = A$, iff $B \subseteq A$ (*iv*) $A \cap B = A$, iff $A \subseteq B$.

Proof (*i*) To prove $A \cup A = A$.

Here $x \in A \cup A \Rightarrow x \in A$ or $x \in A$

$\Rightarrow$ $x \in A$

$\therefore$ $A \cup A \subseteq A$ ...(α)

Conversely, $x \in A \Rightarrow x \in A$ or $x \in A$

$\Rightarrow$ $x \in A \cup A$

$\therefore$ $A \subseteq A \cup A$ ...(β)

$\therefore$ from eqn. (α) and (β) $\Rightarrow$ $A \cup A = A$.

(*ii*) To prove $A \cap A = A$

Here $x \in A \cap A \Rightarrow x \in A$ and $x \in A$

$\Rightarrow$ $x \in A$

$\therefore$ $A \cap A \subseteq A$ ...(γ)

Conversely, $x \in A \Rightarrow x \in A$

$\Rightarrow$ $x \in A \cap A$

$\therefore$ $A \subseteq A \cap A$ ...(δ)

$\therefore$ from eqn. (γ) and (δ) $\Rightarrow$ $A \cap A = A$.

(*iii*) To prove $A \cup B = A$, iff $B \subseteq A$

Let $B \subseteq A$

$\therefore$ $x \in B \Rightarrow x \in A$ ...(*a*)

Now $x \in A \cup B \Rightarrow x \in A$ or $x \in B$

$\Rightarrow$ $x \in A$, by (*a*)

$\therefore$ $A \cup B \subseteq A$ ...(*b*)

Conversely, $x \in A \Rightarrow x \in A$ or $x \in B$

$\Rightarrow$ $x \in A \cup B$

$\therefore$ $A \subseteq A \cup B$ ...(*c*)

Hence, by (*a*) and (*b*), $A \cup B = A$

Now let $A \cup B = A$

$\therefore$ $x \in A$ or $x \in B \Leftrightarrow x \in A$,

i.e., $x \in B \Rightarrow x \in A$

Hence $B \subseteq A$

Thus $A \cup B = A \Leftrightarrow B \subseteq A$.

(*iv*) To prove $A \cap B = A$, if $A \subseteq B$

Let $A \subseteq B$

$\therefore$ $x \in A \Rightarrow x \in B$ ...(*d*)

Now $x \in A \cap B \Rightarrow x \in A$ and $x \in B$

$\Rightarrow$ $x \in A$

$\therefore$ $A \cap B \subseteq A$ ...(*e*)

Also $x \in A \Rightarrow x \in A$ and $x \in B$ by (*d*)

$\Rightarrow$ $x \in A \cap B$

$\therefore$ $A \subseteq A \cap B$

Hence (*e*) and (*f*) $\Rightarrow A \cap B = A$

Now let $A \cap B = A$

Hence $x \in A \Leftrightarrow x \in A$ and $x \in B$

$\therefore$ $x \in A \Rightarrow x \in B$

$\therefore$ $A \subseteq B$

Hence $A \cap B = A \Leftrightarrow A \subseteq B$.

Deductions. (*i*) $A \cup U = U$, $\because$ $A \subseteq U$.

(*ii*) $A \cap U = A$, $\because$ $A \subseteq U$.

(*iii*) $A \cup \phi = A$, $\because$ $\phi \subset A$.

(*iv*) $A \cap \phi = \phi$, $\because$ $\phi \subset A$.

1.8 Properties of Set Operations

1.8.1 Properties of Intersection on Sets

(*i*) *Intersection operation is commutative :* $A \cap B = B \cap A$.

Proof. Let $x \in A \cap B$

But $x \in A \cap B \Rightarrow x \in A$ and $x \in B$ (by definition)

$\Rightarrow$ $x \in B$ and $x \in A$

$\Rightarrow$ $x \in B \cap A$

$\therefore$ $A \cap B \subseteq B \cap A$ ...(*i*)

Conversely let $y \in B \subseteq A$

But $y \in B \cap A \Rightarrow y \in B$ and $y \in A$

$\Rightarrow$ $y \in A$ and $y \in B$

$\therefore$ $B \cap A \subseteq A \cap B$ ...(*ii*)

Hence by (*i*) and (*ii*), we conclude $A \cap B = B \cap A$.

(*ii*) *Associative law holds in case of intersection of three sets A, B, C*: $(A \cap B) \cap C = A \cap (B \cap C)$.

Proof. Let $x \in (A \cap B) \cap C$

$\Rightarrow$ $x \in A \cap B$ and $x \in C$

$\Rightarrow$ $x \in A, x \in B$ and $x \in C$

$\Rightarrow$ $x \in A$ and $x \in B, x \in C$

$\Rightarrow$ $x \in A, x \in B \cap C$

$\Rightarrow$ $x \in A \cap (B \cap C)$

$\therefore$ $(A \cap B) \cap C \subseteq A \cap (B \cap C)$ ...(*i*)

Conversely, let $y \in A \cap (B \cap C)$

But $y \in A \cap (B \cap C)$

$\Rightarrow$ $y \in A$ and $y \in B \cap C$

$\Rightarrow$ $y \in A$, and $y \in B, y \in C$

$\Rightarrow$ $y \in A \cap B$ and $y \in C$

$\Rightarrow$ $y \in (A \cap B) \cap C$

$\therefore$ $A \cap (B \cap C) \subseteq (A \cap B) \cap C$ ...(*ii*)

Hence by (*i*) and (*ii*), we get

$(A \cap B) \cap C = A \cap (B \cap C)$.

1.8.2 Properties of Union on Sets

(*i*) *Commutative law. To prove* $A \cup B = B \cup A$.

Proof. Let $x \in A \cup B$

$x \in A \cup B$

$\Rightarrow$ $x \in A$ or $x \in B$

$\Rightarrow$ $x \in B$ or $x \in A$

$\Rightarrow \quad x \in B \cup A$

$\therefore \quad A \cup B \subseteq B \cup A \quad ...(i)$

Conversely, let $y \in B \cup A$

But $y \in B \cup A$

$\Rightarrow \quad y \in B \text{ or } y \in A$

$\Rightarrow \quad y \in A \text{ or } y \in B$

$\Rightarrow \quad y \in A \cup B$

$\therefore \quad B \cup A \subseteq A \cup B \quad ...(ii)$

Hence by (*i*) and (*ii*), we get

$$A \cup B = B \cup A.$$

(*ii*) *Associative law of union operation. To prove* $A \cup (B \cup C) = (A \cup B) \cup C$.

Proof. Let $x \in A \cup (B \cup C)$

But $x \in A \cup (B \cup C)$

$\Rightarrow \quad x \in A \text{ or } x \in B \cup C$

$\Rightarrow \quad x \in A \text{ or } (x \in B \text{ or } x \in C)$

$\Rightarrow \quad (x \in A \text{ or } x \in B) \text{ or } x \in C$

$\Rightarrow \quad x \in A \cup B \text{ or } x \in C$

$\Rightarrow \quad x \in (A \cup B) \cup C$

$\therefore \quad A \cup (B \cup C) \subseteq (A \cup B) \cup C \quad ...(i)$

Conversely, let $y \in (A \cup B) \cup C$

But $y \in (A \cup B) \cup C$

$\Rightarrow \quad y \in A \cup B \text{ or } y \in C$

$\Rightarrow \quad (y \in A \text{ or } y \in B) \text{ or } y \in C$

$\Rightarrow \quad y \in A \text{ or } (y \in B \text{ or } y \in C)$

$\Rightarrow \quad y \in A \text{ or } y \in B \cup C$

$\Rightarrow \quad y \in A \cup (B \cup C)$

$\therefore \quad (A \cup B) \cup C \subseteq A \cup (B \cup C) \quad ...(ii)$

Hence by (*i*) and (*ii*), we have

$$A \cup (B \cup C) = (A \cup B) \cup C.$$

(*iii*) *Distributive laws: For three given sets A, B, C, to prove*

(*a*) $A \cap (B \cap C) = (A \cap B) \cup (A \cap C)$ and

(*b*) $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$.

Proof. (*a*) Let $x \in A \cap (B \cup C)$

But $x \in A \cap (B \cup C)$

$\Rightarrow$ $x \in A$ and $x \in B \cup C$

$\Rightarrow$ $x \in A$ and $(x \in B$ or $x \in C)$

$\Rightarrow$ $(x \in A$ and $x \in B)$ or $(x \in A$ and $x \in C)$

$\Rightarrow$ $x \in A \cap B$ or $x \in A \cap C$

$\Rightarrow$ $x \in (A \cap B) \cup (A \cap C)$

Hence $A \cap (B \cup C) \subseteq (A \cap B) \cup (A \cap C)$ ...(*i*)

Conversely, let $y \in (A \cap B) \cup (A \cap C)$

But $y \in (A \cap B) \cup (A \cap C)$

$\Rightarrow$ $y \in A \cap B$ or $y \in A \cap C$

$\Rightarrow$ $(y \in A, y \in B)$ or $(y \in A$ or $y \in C)$

$\Rightarrow$ $y \in A$ and $[y \in B$ or $y \in C]$

$\Rightarrow$ $y \in A$ and $y \in B \cup C$

$\Rightarrow$ $y \in A \cap (B \cup C)$

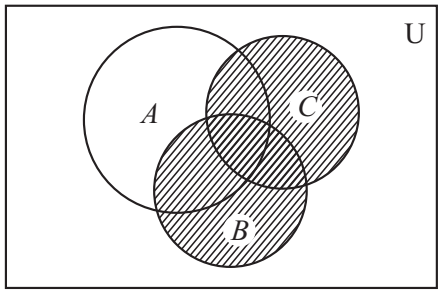
$\therefore (A \cap B) \cup (A \cap C) \subseteq A \cap (B \cup C)$ ...(*ii*)

Hence by (*i*) and (*ii*), we conclude

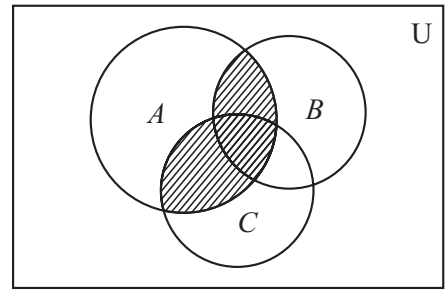
$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

This is also expressed by saying that intersection of sets is distributive over union of sets.

Venn-diagrams for this theorem are given below:



$B \cup C$



$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

Fig. 1.4

(*b*) Let $x \in A \cup (B \cap C)$

But $x \in A \cup (B \cap C)$

$\Rightarrow$ $x \in A$ or $x \in B \cap C$

$\Rightarrow$ $x \in A$ or $[x \in B$ and $x \in C]$

$\Rightarrow$ $[x \in A$ or $x \in B]$ and $[x \in A$ or $x \in C]$

$\Rightarrow$ $x \in A \cup B$ and $x \in A \cup C$

$\Rightarrow$ $x \in (A \cup B) \cap (A \cup C)$

$\therefore$ $A \cup (B \cap C) \subseteq (A \cup B) \cap (A \cup C)$ ...(*i*)

Conversely, let $y \in (A \cup B) \cap (A \cup C)$

But $y \in (A \cup B) \cap (A \cup C)$

$\Rightarrow$ $y \in A \cup B$ and $y \in A \cup C$

$\Rightarrow$ $(y \in A$ or $y \in B)$ and $(y \in A$ or $y \in C)$

$\Rightarrow$ $y \in A$ or $(y \in B$ and $y \in C)$

$\Rightarrow$ $y \in A$ or $y \in B \cap C$

$\Rightarrow$ $y \in A \cup (B \cap C)$

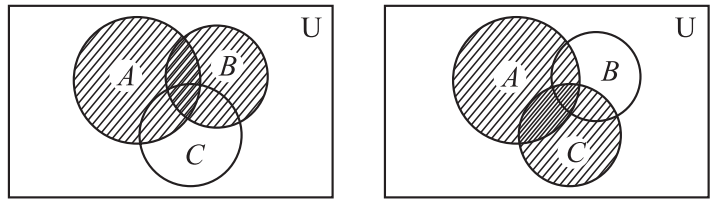
$\therefore$ $(A \cup B) \cap (A \cup C) \subseteq A \cup (B \cap C)$ ...(*ii*)

Hence by (*i*) and (*ii*), we have

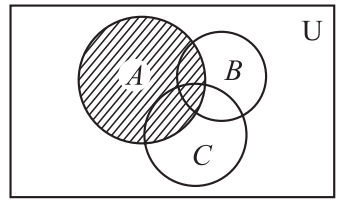
$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

This is also expressed by saying that union of sets is distributive over intersection of sets.

Venn-diagrams of this theorem are as follows:



$A \cup B$ $A \cup C$



$(A \cup B) \cap (A \cup C) = A \cup (B \cap C)$

Fig. 1.5

1.8.3 Number of Elements in a Union of Two or more Sets

Let A, B, C be three finite sets and let $n(A)$, $n(B)$, $n(C)$, respectively, denote the number of elements in these sets. Then, we see that

$$n(A \cup B) = n(A) + n(B) - n(A \cap B) \qquad ...(i)$$

In case A, B are disjoint sets, then

$$A \cap B = \phi \text{ and } n(A \cap B) = n(\phi) = 0$$

$\therefore$ For disjoint sets A and B,

$$n(A \cup B) = n(A) + n(B)$$

Furthermore, it can be verified that

$$n(A \cup B \cup C) = n(A) + n(B) + n(C) - n(A \cap B) - n(B \cap C) - n(A \cap C) + n(A \cap B \cap C) \qquad ...(ii)$$

In problems, involving the use of formulae (i) and (ii), sometimes it becomes imperative to expand a set A with reference to a set B. In that case, we use the result.

$$A = (A \cap B) \cup (A \cap B') \qquad ...(iii)$$

It follows from (iii) that

$$A \cap B = (A \cap B \cap C) \cup (A \cap B \cap C') \qquad ...(iv)$$

In more complicated problems on finding the number of elements in a set, it is advisable to use Venn-diagrams.

Theorem. Prove that $A - B = A \cap B' = B' - A'$.

Proof. Let $x \in A - B$

$\therefore$ $x \in A - B$

$\Rightarrow$ $x \in A$ and $x \notin B$

$\Rightarrow$ $x \in A$ and $x \in B'$

$\Rightarrow$ $x \in A \cap B'$

$\therefore$ $A - B \subseteq A \cap B'$ ...(i)

Conversely, $x \in A \cap B'$

$\Rightarrow$ $x \in A, x \in B'$

$\Rightarrow$ $x \in A, x \notin B$

$\Rightarrow \quad x \in A - B$

$\therefore \quad A \cap B \subseteq A - B' \quad ...(ii)$

(i) from eqn. $(ii) \quad \Rightarrow (A - B) = A \cap B'$

Now $\quad B' - A' = B' \cap (A')'$ by 1st part

$= B' \cap A \quad \because \ (A')' = A$

$= A \cap B'$

Hence $\quad A - B = A \cap B' = B' - A'.$

Augustus De Morgan (1806–1871)

Augustus De Morgan (1806–1871) was born in Madurai, India, where his father was a colonel in the Indian army. When the young De Morgan was 7 months old, the family moved to England. He attanded private schools, where he studied Latin, Greek and Hebrew to develop a strong interest in Mathematics. He completed his graduation from Trinity College in 1827 and pondered career either in medicine or law, but pursued mthematics. His professional career began in 1828 at University College London. Due to some problem he return to Trinity in 1836 where he worked upto 1866. He was a fellow of Astronomical society and founder of London Mathematical society. De Morgan authored more than 100 articles for more than 15 journals as well as a number of text books.

In 1838 he coined the term mathematical Induction and gave a clear justification to this proof method, although it had been in use. His the Differential and Integral Calculus (1842) gives the first precise definition of a limit and some tests for convergence of infinite series. He had a keen interest in History of mathematics. He wrote biographies of Sir Isaac Newton and Edmund Halley. His wife wrote De Morgan's biography in 1882. His researches shows social structure problems.

1.9 De-Morgan's Laws

To prove that

$(a) \ (A \cup B)' = A' \cap B'$

$(b) \ (A \cap B)' = A' \cup B'$

$(c) \ A - (B \cup C) = (A - B) \cap (A - C)$

$(d) \ A - (B \cap C) = (A - B) \cup (A - C)$

Proof. (*a*) Let $x \in (A \cup B)'$

But $x \in (A \cup B)'$

$\Rightarrow$ $x \notin A \cup B$

$\Rightarrow$ $x \notin A$ and $x \notin B$

$\Rightarrow$ $x \in A'$ and $x \in B'$

$\Rightarrow$ $x \in A' \cap B'$

$\therefore$ $(A \cup B)' \subseteq A' \cap B'$ ...(*i*)

Conversely, let $y \in A' \cap B'$

But $y \in A' \cap B'$

$\Rightarrow$ $y \in A'$ or $y \in B'$

$\Rightarrow$ $y \notin A$ or $y \notin B$

$\Rightarrow$ $y \notin A \cup B$

$\Rightarrow$ $y \in (A \cap B)'$

$\therefore$ $A' \cap B' \subseteq (A \cup B)'$ ...(*ii*)

Hence by (*i*) and (*ii*), we get

$$(A \cup B)' = A' \cap B'.$$

(*b*) Let $x \in (A \cap B)'$

But $x \in (A \cap B)'$

$\Rightarrow$ $x \notin A \cap B$

$\Rightarrow$ $x \notin A$ or $x \notin B$ (Note)

$\Rightarrow$ $x \in A'$ or $x \in B'$

$\Rightarrow$ $x \notin A' \cup B'$ ...(*i*)

Conversely, let $(y \in A' \cup B')$

But $y \in A' \cup B'$

$\Rightarrow$ $y \in A'$ or $y \in B'$

$\Rightarrow$ $y \notin A$ or $y \notin B$

$\Rightarrow$ $y \notin A \cup B$

$\Rightarrow$ $y \in (A \cup B)'$

$\therefore$ $A' \cap B' \subseteq (A \cap B)'$ ...(*ii*)

Hence by (*i*) and (*ii*), we get

$$(A \cap B)' = A' \cup B'.$$

(*c*) Let $x \in A - (B \cup C)$

But $x \in A - (B \cup C)$

$\Rightarrow$ $x \in A$ and $x \notin B \cup C$

$\Rightarrow$ $x \in A$ and $(x \notin B$ and $x \notin C)$

$\Rightarrow$ $(x \in A$ and $x \notin B)$ and $(x \in A$ and $x \notin C)$

$\Rightarrow$ $x \in (A - B)$ and $x \in (A - C)$

Hence $A - (B \cup C) \subseteq (A - B) \cap (A - C)$ ...(*i*)

Conversely, let $y \in (A - B) \cap (A - C)$

But $y \in (A - B) \cap (A - C)$

$\Rightarrow$ $y \in (A - B)$ and $y \in (A - C)$

$\Rightarrow$ $y \in A$ but $y \notin B$ and $y \in A$ but $y \notin C$

$\Rightarrow$ $y \in A$ but $y \notin B, y \notin C$

$\Rightarrow$ $y \in A$ but $y \notin B \cup C$ (Note)

$\Rightarrow$ $y \in A - (B \cup C)$

$\therefore$ $(A - B) \cap (A - C) \subseteq A - (B \cup C)$ ...(*ii*)

Thus by (*i*) and (*ii*), we get

$$A - (B \cup C) = (A - B) \cap (A - C)$$

(*d*) Here $A - (B \cap C) = A \cap (B \cap C)'$

$= A \cap (B' \cap C')$ by (*b*)

$= (A \cap B') \cup (A \cap C')$ by distributive law

$= (A - B) \cup (A - C)$.

Theorem 1. Let A and B be two finite sets. Then

$$| A \cup B | = | A | + | B | - | A \cap B |$$

Proof. We use Venn diagram

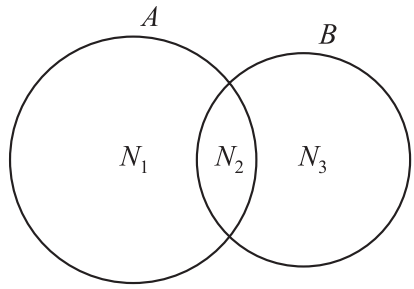


Fig. 1.6

Let N_1 represent the number of elements present exclusively in A. *i.e.*, $N_1 = |A|$

N_2 represent the number of elements present in both A and B. *i.e.*, $N_2 = |A \cap B|$

N_3 represent the number of elements present exclusively in B. *i.e.*, $N_3 = |B|$.

Then, $$|A \cup B| = N_1 + N_2 + N_3$$
$$= (N_1 + N_2) + (N_2 + N_3) - N_2$$
$$|A \cup B| = |A| + |B| - |A \cap B|.$$

Theorem 2. Extension of theorem 1. If A, B and C are three finite sets, then

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C|$$

Proof. Let $B \cup C = D$, then

$$|A \cup B \cup C| = |A \cup D|$$
$$= |A| + |D| - |A \cap D| \quad \text{(using theorem 1)} \quad ...(i)$$

But $$|D| = |B \cup C| = |B| + |C| - |B \cap C| \quad ...(ii)$$

$\therefore$ $$|A \cap D| = |A \cap (B \cup C)| = |(A \cap B) \cup (A \cap C)|$$
$$= |A \cap B| + |A \cap C| - |(A \cap B) \cup (A \cap C)|$$
$$= |A \cap B| + |A \cap C| - |A \cap B \cap C| \quad ...(iii)$$

Using (*ii*) and (*iii*) in (*i*), we get

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C|.$$

Venn diagram for theorem 2

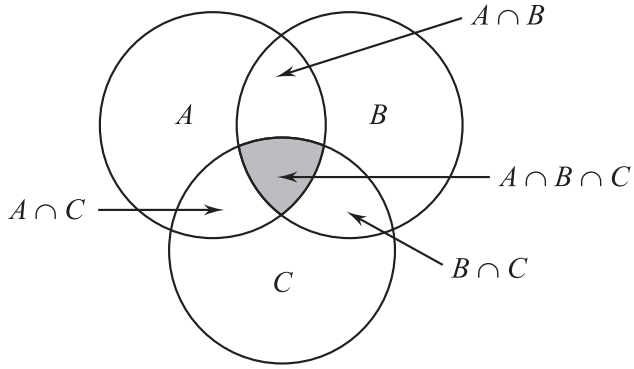


Fig. 1.7

Note: We may extend the theorem for four finite sets A, B, C and D also *i.e.*,
$$|A \cap B \cap C \cap D| = |A| + |B| + |C| + |D| - |A \cap B| - |A \cap C| - |A \cap D| - |B \cap C| - |B \cap D| - |C \cap D| + |A \cap B \cap C| + |A \cap B \cap D| + |A \cap C \cap D| + |B \cap C \cap D| - |A \cap B \cap C \cap D|.$$

1.10 General form of Principle of Inclusion and Exclusion

If $A_1, A_2, A_3, ..., A_n$ are 'n' finite sets, then

$$|A_1 \cup A_2 \cup A_3 \cup ..., \cup A_n| = \sum_{1 \le i \le n} |A_i| - \sum_{1 \le i \le j \le n} |A_i \cap A_j| + \sum_{1 \le i \le j \le k \le n} |A_i \cap A_j \cap A_k| - \sum_{1 \le i \le j \le k \le l \le n} |A_i \cap A_j \cap A_k \cap A_l| + \Sigma ... + (-1)^{n-1} |A_1 \cap A_2 \cap A_3 ... \cap A_n|$$

Example 11

Verify the principle of inclusion and exclusion for the sets

$A = \{1, 2, 3, 4, 5, 6\}$, $B = \{2, 4, 7, 8, 9\}$ *and* $C = \{1, 2, 4, 7, 10, 12\}$

Solution

We have $A \cup B \cup C = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12\}$

$A \cap B = \{2, 4\}$, $B \cap C = \{2, 4, 7\}$, $A \cap C = \{1, 2, 4\}$,

$[A \cap B \cap C] = \{2, 4\}$.

Also

$|A \cup B \cup C| = 11$, $|A| = 6$, $|B| = 5$, $|C| = 6$

$|A \cap B \cap C| = 2$, $|A \cap B| = 2$, $|B \cap C| = 3$, $|A \cap C| = 3$

The principle of inclusion and exclusion is

$$|A \cup B \cup C| = 11, \ |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$$

$$= 6 + 5 + 6 - 2 - 3 - 3 + 2 = 11$$

Hence the theorem is verified. ■

Example 12

In a survey of 1000 *households, washing machines, vacuum cleaners and refrigerators were counted. Each house had at least one of these appliances,* 400 *had no refrigerator,* 380 *no vacuum cleaner and* 542 *no washing machine,* 294 *had both a vacuum cleaner and a washing machine,* 277 *both a refrigerator and a vacuum cleaner,* 190 *both a refrigerator and a washing machine. How many households had all the three appliances? How may had only a vacuum cleaner?*

Solution

Let A, B, C denote the sets of households having a washing machine, a vacuum cleaner and a refrigerator respectively.

Hence, we are given that

$$n(A \cup B \cup C) = 1000;\ n(C') = 400;$$
$$n(B') = 380;\ n(A') = 542$$
$$n(A \cap B) = 294;\ n(B \cap C) = 277;\ n(A \cap C) = 190$$

Also $A \cup B \cup C = U$ = universal set

and $n(A \cup B \cup C) = n(U) = 1000$

But $C \cap C' = \phi$ and $n(C \cap C') = 0$

$\therefore$ $n(C) + n(C') = n(U)$ gives $1000 = n(C) + 400$ or $n(C) = 600$

Similarly $n(A) = 458$ and $n(B) = 620$

$\therefore$ From $n(A \cup B \cup C) = n(A) + n(B) + n(C) - n(A \cap B)$
$$- n(A \cap C) - n(B \cap C) + n(A \cap B \cap C)$$
$$1000 = 458 + 620 + 600 - 294 - 190 - 277 + n(A \cap B \cap C)$$

or $n(A \cap B \cap C) = 83.$

This gives the number of households having all the appliances.

Now $B \cap C' \cap A'$ is set of all households which have a vacuum cleaner but no washing machine or refrigerator. Hence we are to find out $n(B \cap C' \cap A')$.

But $(B \cap C' \cap A') \cup (B' \cap C' \cap A') = C' \cap A'$ by formula.

But there is no household which has no appliances, *i.e.*,

$$n(B' \cap C' \cap A') = 0$$

$\therefore$ $n(B \cap C' \cap A') = n(C' \cap A') = n(U) - n(A \cup C)$
$$= n(U) - \{n(A) + n(C) - n(A \cap C)\}$$
$$= 1000 - 458 - 600 + 190 = 132.$$

This is the number of households having refrigerator only. ■

Example 13

Seventy-five children went to an amusement park where they can ride on the merry-go-round, roller coaster and Ferrywheel. It is known that 20 *of them took all the rides and* 55 *have taken at least* 2 *of the three rides. Each ride costs* ₹ 0.5 *and the total receipt of park was* ₹ 70. *Determine the number of children who did not try any of the rides.*

Solution

Let A by the set of children who have the ride on Merry-go-round,

B be the set of children who have the ride on Roller,

C be the set of children who have the ride an Ferry wheel.

Then $|A \cap B \cap C| = 20$ = Number of children who took all the rides.

Total number of rides $= 70 \times 2 = 140 = |A| + |B| + |C|$

Number of children who took atleast 2 rides

$$= 55 = |(A \cap B) \cup (B \cap C) \cup (A \cap C)|$$

But by the Principle of addition, we have

$$|(A \cap B) \cup (A \cap C) \cup (B \cap C)| = |A \cap B| + |A \cap C| + |B \cap C| - |(A \cap B) \cap (A \cap C)| - |(A \cap C) \cap (B \cap C)| - |(A \cap B) \cap (B \cap C)| + |(A \cap B) \cap (A \cap C) \cap (B \cap C)|$$

$$= |A \cap B| + |B \cap C| + |A \cap C| - |A \cap B \cap C| - |A \cap B \cap C| - |A \cap B \cap C| + |A \cap B \cap C|$$

$$= |A \cap B| + |B \cap C| + |A \cap C| - 2|A \cap B \cap C|$$

i.e., $|A \cap B| + |B \cap C| + |A \cap C| - 2 \times 20 = 55$

i.e., $|A \cap B| + |B \cap C| + |A \cap C| = 95$

Also $|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C|$

$= 140 - 95 + 20$

$= 65$ (No. of children who took at least a ride)

$\therefore$ No. of children who did not take any ride $= 75 - 65 = 10$. ■

Example 14

A survey shows that 58% *of Indians like coffee where as* 75% *like tea. What percentage of Indians like both coffee and tea?*

Solution

Let $A \rightarrow$ % of Indians who drink coffee

$B \rightarrow$ % of Indians who drink tea

$\therefore$ $A \cap B \rightarrow$ % of Indians who drink both coffee and tea.

$$|A| = 58, \ |B| = 75 \ \text{ and } \ |A \cup B| = 100$$

To find $|A \cap B|$

We know that $|A \cup B| = |A| + |B| - |A \cap B|$

$$100 = 58 + 75 - |A \cap B|$$

i.e., $$|\mathrm{A} \cap \mathrm{B}| = 133 - 100 = 33$$

$\therefore$ 33% of Indians drink both coffee and tea. ■

Example 15

A computer company requires 30 *programmers to handle systems programming jobs and* 40 *programmers for applications programming. If the company appoints* 55 *programmers to carry out these jobs, how many of these perform jobs of both types?*

Solution

A denote the set of programmers who handle systems programming job and B the set of programmers who handle applications programming. Then $A \cup B$ is the set of programmers appointed to carry out these jobs. We have from given condition.

$$|A| = 30, |B| = 40, |A \cup B| = 55$$

Using additive rule, we get

$$|A \cup B| = |A| + |B| - |A \cap B|$$

gives $$|A \cap B| = |A| + |B| - |A \cup B|$$

$$= 30 + 40 - 55 = 15.$$

This means that 15 programmers perform both types of jobs. ■

Example 16

A survey of 500 *television watchers produced the following information*: 285 *watch football,* 195 *watch hockey,* 115 *watch basketball,* 45 *watch football and basketball,* 70 *watch foot ball and hockey,* 50 *watch hockey and basketball and* 50 *do not watch any of the three kinds of games.*

(*a*) *How many people in the survey watch all three kinds of games?*

(*b*) *How many people watch exactly one of the sports?*

Solution

Let U denote the set of all people included in the survey, A denote the set of people who watch football, B denote the set of people who watch hockey, and C denote the set of people who watch basketball. Then, from what is given, we have

$$|U| = 500,\ |A| = 285,\ |B| = 195,\ |C| = 115$$

$$|A \cap C| = 45,\ |A \cap B| = 70,\ |B \cap C| = 50$$

$$|A \cup B \cup C| = 50,\ |A \cup B \cup C| = 500 - 50 = 450$$

Using the addition rule we get

$$|A \cap B \cap C| = |A \cup B \cup C| - |A| - |B| - |C| + |A \cap B| + B \cap C| + |A \cap C|$$

$$= 450 - 285 - 195 - 115 + 70 + 50 + 45 = 20$$

The number of people who watch all three kinds of games is 20.

Let A_1 denote the set of people who watch *only* football, B_1 denote the set of people who watch *only* hockey and C_1 denote the set of people who watch *only* basketball.

Then, $A_1 = A - B - C$, and by virtue of the result,

$$|A_1| = |A| - |A \cap B| - |A \cap C| + |A \cap B \cap C|$$

Accordingly, the number of people who watch only football is

$$|A_1| = 285 - 70 - 45 + 20 = 190$$

Similarly, the number of people who watch only hockey is

$$|B_1| = |B| - |B \cap A| - |B \cap C| + |B \cap C \cap A|$$

$$= 195 - 70 - 50 + 20 = 95$$

and the number of people who watch only basketball is

$$|C_1| = |C| - |C \cap A| - |C \cap B| + |C \cap A \cap B|$$
$$= 115 - 45 - 50 + 20 = 40$$

From these, we find that the number of people who watch exactly one of the sports is

$$|A_1| + |B_1| + |C_1| = 325.$$ ■

Example 17

It is known that at a university 60% *of the professors play tennis,* 50% *of them play bridge,* 70% *jog,* 20% *play tennis and bridge,* 30% *play tennis and jog and* 40% *play bridge and jog. If someone claimed that* 20% *of the professors play all the* 3 *games, would you believe this claim? why*?

Solution

Let A: set of professors who play Tennis
B: set of professors who play bridge
C: set of professors who play jog.

$$\therefore \quad |A| = 60,\ |B| = 50,\ |C| = 70$$
$$|A \cap B| = 20,\ |A \cap C| = 30,\ |B \cap C| = 40$$
$$|A \cup B \cup C| = 100$$

But $|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$

$$100 = 60 + 50 + 70 - 20 - 30 - 40 + |A \cap B \cap C|$$

i.e., $|A \cap B \cap C| = 10$

i.e., 10% of the professors play all the 3 games.

But it is given that $|A \cap B \cap C| = 20$

Hence the claim is false. ■

Example 18

In a class of 100 *students,* 39 *play Tennis,* 58 *play Cricket,* 32 *play Hockey*, 10 *play Cricket and Hockey,* 11 *play Hockey and Tennis,* 13 *play Tennis and Cricket. How many students play*

(*a*) *all the* 3 *games*
(*b*) *just one game and*
(*c*) *Tennis and Cricket and not Hockey?*

Solution

Let S be the set of all students

T : set of students who play Tennis

C : set of students who play Cricket

H : set of students who play Hockey.

Then $T \cap C$: set of students who play Tennis and Cricket

$T \cap H$: set of students who play Tennis and Hockey

$C \cap H$: set of students who play Cricket and Hockey

$\therefore \quad |S| = |T \cup C \cup H| = 100$

(a) We have by the principle of Inclusion and exclusion

$$|T \cup C \cup H| = |T| + |C| + |H| - |T \cap C| - |T \cap H| - |C \cap H| + |T \cap C \cap H|$$

i.e., $$100 = 39 + 58 + 32 - 13 - 11 - 10 + |T \cap C \cap H|$$

or $$|T \cap C \cap H| = 5$$

= No. of students who play all the games.

(b) No. of students who play only Cricket

$$= |C| - |T \cap C| - |C \cap H| + |C \cap H \cap T|$$

$$= 58 - 13 - 10 + 5 = 40$$

No. of students who play only Tennis

$$= |H| - |T \cap C| - |T \cap H| + |T \cap C \cap H|$$

$$= 39 - 13 - 11 + 5 = 20$$

No. of students who play only Hockey

$$= |H| - |T \cap H| - |T \cap C| + |T \cap C \cap H|$$

$$= 32 - 11 - 13 + 5 = 13$$

$\therefore$ No. of students who just play one game

$$= 40 + 20 + 13 = 73$$

(c) No. of students who play Tennis and Cricket But not Hockey

$$= |T \cap C| - |T \cap C \cap H|$$

$$= 13 - 5 = 8.$$ ■

Example 19

A survey on a sample of 25 *new cars being sold by an auto dealer was conducted to see which of the three popular options: air conditioning, radio and power windows, were already installed. The survey found:* 15 *had air-conditioning,* 12 *had radio,* 11 *had power windows,* 5 *had air-conditioning and power windows,* 9 *had air-conditioning and radio,* 4 *had radio and power windows, and* 3 *had all three options. Find the number of cars that had:*

(*i*) *only power windows,*
(*ii*) *only air-conditioning,*
(*iii*) *only radio,*
(*iv*) *only one of the options,*
(*v*) *at least one option,*
(*vi*) *none of the options.*

Solution

Let A, R and W be the sets of cars included in the sample that had air-conditioning, radio and power windows, respectively. Also, let U denote the set of all cars in the sample. Then from what is given we have the following:

$$|U| = 25,\ |A| = 15,\ |R| = 12,\ |W| = 11,\ |A \cap W| = 5$$
$$|A \cap R| = 9,\ |R \cap W| = 4,\ |A \cap R \cap W| = 3$$

Now, let W_1, A_1 and R_1 respectively denote the sets of cars that had only power windows, only air-conditioning, and only radio. Then we find that

$$\begin{aligned} |W_1| &= |W - A - R| \\ &= |W| - |W \cap A| - |W \cap R| + |W \cap A \cap R| \\ &= 11 - 5 - 4 + 3 = 5 \end{aligned}$$

Thus, the number of cars that had only power windows is 5.

Similarly, the number of cars which had only air-conditioning is

$$\begin{aligned} |A_1| &= |A| - |A \cap W| - |A \cap R| + |A \cap W \cap R| \\ &= 15 - 5 - 9 + 3 = 4 \end{aligned}$$

and the number of cars which had only radio is

$$\begin{aligned} |R_1| &= |R| - |R \cap W| - |R \cap A| + |R \cap W \cap A| \\ &= 12 - 4 - 9 + 3 = 2 \end{aligned}$$

Consequently, the number of cars that had only one of the options is

$$| W_1 \cup A_1 \cup R_1 | = | W_1 | + | A_1 | + | R_1 |$$
$$= 5 + 4 + 2 = 11$$

Next, the number of cars which had at least one option is

$$| W \cup A \cup R | = | W | + | A | + | R | - | W \cap A | - | W \cap R |$$
$$- | A \cap R | + | W \cap A \cap R |$$
$$= 11 + 15 + 12 - 5 - 9 - 4 + 3 = 23$$

Lastly, the number of cars that had none of the options is

$$| U | - | W \cup A \cup R | = 25 - 23 = 2.$$ ■

Example 20

Let X be the set of all three-digit integers; that is,

$$X = \{x \text{ is an integer} : 100 \le x \le 999\}$$

If A_i is the set of numbers in X whose ith digit is i, compute the cardinality of the set $A_1 \cup A_2 \cup A_3$.

Solution

We first observe the following:

$A_1 = \{100, 101, 102, \ldots 199\}$ so that $| A_1 | = 100$

$A_2 = \{120, 121, 122, \ldots 129, 220, 221, \ldots 229, \ldots 320, 321, \ldots 329,$
$\ldots 920, 921, 922, \ldots 929\}$ so that $| A_2 | = 90$, and

$A_3 = \{103, 113, 123, \ldots 193, 203, 213, 223, \ldots, 293, 303, 313, 323,$
$\ldots 393, \ldots 903, 913, 923, \ldots 993\}$

so that $| A_3 | = 90$

Further, we find that

$A_1 \cap A_2 = \{120, 121, 122, \ldots 129\}$ so that $| A_1 \cap A_2 | = 10$

$A_1 \cap A_3 = \{103, 113, 123, \ldots 193\}$ so that $| A_1 \cap A_3 | = 10$

$A_2 \cap A_3 = \{123, 223, 323, \ldots 923\}$ so that $| A_2 \cap A_3 | = 9$

$A_1 \cap A_2 \cap A_3 = \{123\}$ so that $| A_1 \cap A_2 \cap A_3 | = 1$

Therefore,

$$| A_1 \cup A_2 \cup A_3 | = | A_1 | + | A_2 | + | A_3 | - | A_1 \cap A_2 | - | A_2 \cap A_3 |$$
$$- | A_3 \cap A_1 | + | A_1 \cap A_2 \cap A_3 |$$
$$= 100 + 90 + 90 - 10 - 10 - 9 + 1 = 252.$$ ■

Example 21

Prove that:

(*i*) $A \oplus (A \cap B) = A - B$

(*ii*) $A \cap (B \oplus C) = (A \cap B) \oplus (A \cap C)$

Solution

Here, by definition

$$A \oplus (A \cap B) = [A - (A \cap B)] \cup [(A \cap B) - A]$$
$$= (A - B) \cup \phi$$

for there is no element in $A \cap B$ which does not belong to A

i.e., $(A \cap B) - A = \phi$

Also $(A - B) \cup \phi = A - B$

$\therefore$ $A \oplus (A \cap B) = A - B$

(*ii*) Here $A \cap (B \cap C) = A \cap [(B - C) \cup (C - B)]$

$= [A \cap (B - C)] \cup [A \cap (C - B)]$ [by distributive law]

$= [(A \cap B) - (A \cap C)] \cup [(A \cap C) - (A \cap B)]$

$= (A \cap B) \oplus (A \cap C)$ (by definition.) ■

Example 22

Prove that $(A \oplus B) \oplus (A \cap B) = A \cup B$.

Solution

Here $(A \oplus B) \oplus (A \cap B)$

$= [(A - B) \cup (B - A)] \oplus (A \cap B)$

$= [(A - B) \cup (B - A) - A \cap B] \cup [(A \cap B) - (A - B) \cup (B - A)]$ (by definition)

$= [\{(A \cup B) - (A \cap B)\} - A \cap B] \cup [A \cap B - \{(A \cup B) - (A \cap B)\}]$

But $[(A \cup B) - (A \cap B)] - (A \cap B) = (A \cup B) - (A \cap B)$

Also $A \cap B$ has no element in common with $(A \cup B) - (A \cap B)$

Hence $A \cap B - \{(A \cup B) - (A \cap B)\} = A \cap B$

$\therefore$ $(A \oplus B) \oplus (A \cap B) = [(A \cup B) - (A \cap B)] \cup [A \cap B]$

$= A \cup B.$ ■

Example 23

If $A = \{a, b, c\}$, $B = \{a, c, m, n\}$, $C = \{c, l, m, n, q\}$. *Verify the following laws:*

(*a*) $(A \cap B) \cup C = (A \cup C) \cap (B \cup C)$

(*b*) $(A - B) \neq (B - A)$

(*c*) $A - (B \cap C) = (A - B) \cup (A - C)$

(*d*) $(A \cup B)' = A' \cap B'$

if set of all letters of English alphabet is the universal set.

Solution

(*a*) By definition $A \cap B$ = set of all elements common to A and B

$$= \{a, c\}$$

$\therefore$ $(A \cap B) \cup C = \{a, c\} \cup \{c, l, m, n, q\}$ = set of elements which belong to $A \cap B$ or C

$$= \{a, c, l, m, n, q\}$$

Also $A \cup C = \{a, b, c, l, m, n, q\}$ and $B \cup C = \{a, c, l, m, n, q\}$

$\therefore$ $(A \cup C) \cap (B \cup C)$ = set of elements common to $A \cup C$ and $B \cup C$

$$= \{a, c, l, m, n, q\}$$

Hence $(A \cap B) \cup C = (A \cup C) \cap (B \cup C)$

(*b*) Here $A - B$ = set of elements which belong to A but do no belong to B

$$= \{b\}$$

Similarly $B - A = \{m, e\}$

Hence $A - B \neq B - A$.

(*c*) Here $B \cap C = \{c, m, n\}$ by definition

$\therefore$ $A - (B \cap C)$ = set of elements of A which do not belong to $B \cap C$

$$= \{a, b\}$$

Also $A - B = \{b\}$ and $A - C = \{a, b\}$

$\therefore$ $(A - B) \cup (A - C) = \{a, b\}$

Hence $A - (B \cap C) = (A - B) \cup (A - C)$

(*d*) Here $A \cup B = \{a, b, c, m, n\}$

$\therefore$ $(A \cup B)'$ = set of all letters of the alphabet which do not belong to $A \cup B$

$$= \{d, e, f, g, h, i, j, k, l, o, p, q, r, ..., x, y, z\}$$

But $A' = \{d, e, f, g, h, ..., x, y, z\}$

and $B' = \{b, d, e, f, g, h, i, j, k, l, o, p, q, ... x, y, z\}$

$\therefore$ $A' \cap B' = \{d, e, f, g, h, i, j, k, l, o, p, q, ..., x, y, z\}$

Hence $(A \cup B)' = A' \cap B'$. ■

Example 24

Prove that

(*i*) $A \cap (B - C) = A \cap B - A \cap C$

(*ii*) $(A - B) \cup B = A$ if $B \subset A$

Solution

(*i*) Let $x \in A \cap (B - C)$

$\therefore$ $x \in A \cap (B - C)$

$\Rightarrow$ $x \in A$ and $x \in B - C$

$\Rightarrow$ $x \in A$ and $x \in B$ but $x \notin C$

$\Rightarrow$ ($x \in A$ and $x \in B$) but $x \notin A \cap C$

$\Rightarrow$ $x \in A \cap B$ but $x \notin A \cap C$

$\Rightarrow$ $x \in (A \cap B) - (A \cap C)$

$\therefore$ $A \cap (B - C) \subseteq (A \cap B) - (A \cap C)$ ...(*i*)

Now let $y \in (A \cap B) - (A \cap C)$

$\therefore$ $y \in (A \cap B) - (A \cap C)$

$\Rightarrow$ $y \in A \cap B, y \notin A \cap C$

$\Rightarrow$ $y \in A$ and $y \in B$ and $y \in A$ but $y \notin C$

$\Rightarrow$ $y \in A, y \in B, y \notin C$

$\Rightarrow$ $y \in A, y \in B - C$

$\Rightarrow$ $y \in A \cap (B - C)$

$\therefore$ $(A \cap B) - (A \cap C) \subseteq A \cap (B - C)$ ...(*ii*)

Hence by (*i*) and (*ii*), we get

$$A \cap (B - C) = (A \cap B) - (A \cap C)$$

This may be used as a standard result.

(*ii*) $(A - B) \cup B = A$

$\Rightarrow$ $(A \cap B') \cup B = A$

$\Rightarrow (A \cup B) \cap (B' \cup B) = A$ (by distributive law)

$\Rightarrow \quad (A \cup B) \cap U = A \qquad \because \quad B' \cup B = U$

$\Rightarrow \quad A \cup B = A \qquad \because \quad A \cap U = A$

But $A \cup B = A$

$\Rightarrow$ all elements of A or B are elements of A

$\Rightarrow$ all elements of B are elements of A

$\Rightarrow \quad B \subset A$

$\therefore \quad (A - B) \cup B$

$\Rightarrow \quad B \subseteq A.$ ■

Example 25

Prove the following

(*i*) $(A - B) \cap (B - A) = \phi$

(*ii*) $(A \cup B) \cap B' = A \quad \Leftrightarrow \quad A \cap B = \phi$

Solution

(*i*) Let $A - B = C$ and $B - A = D$

$\therefore \quad (A - B) \cap (B - A) = C \cap D$ ...(*i*)

Now the set C contains no element of B but only elements of A, while D contains no elements of A. Hence C and D have no elements in common

$\therefore \quad C \cap D =$ nul set $= \phi$

$\therefore$ by (*i*) $(A - B) \cap (B - A) = \phi$

(*ii*) Here $(A \cup B) \cap B' = A$

$\Rightarrow (A \cap B') \cup (B \cap B') = A$ by distributive law

$\Rightarrow \quad (A \cap B') \cup \phi = A, \quad \because \quad B \cap B' = \phi$

as B and B' have no common element.

$\therefore \quad (A \cup B) \cap B' = A$

$\Rightarrow \quad (A \cap B') = A$

$\Rightarrow \quad A \subseteq B'$

$\Rightarrow$ A and B have no element in common

$\Rightarrow \quad A \cap B = \phi$ ...(*i*)

Conversely $A \cap B = \phi$

$\Rightarrow$ A and B have no common element

$\Rightarrow \quad A \subseteq B'$

$\Rightarrow \quad A \cap B' = A$

$\Rightarrow \quad (A \cap B') \cup \phi = A$

$\Rightarrow (A \cap B') \cup (B \cap B') = A$

$\Rightarrow \quad (A \cup B) \cap B' = A$ ...(*ii*)

Hence by (*i*) and (*ii*), we say

$(A \cup B) \cap B' = A \quad \Leftrightarrow \quad A \cap B = \phi.$ ■

Example 26

Prove that

(*a*) $(A - B) \cup (B - A) = (A \cup B) - (A \cap B)$ *and*

(*b*) $(A - B) = \phi$ *if* $A \subseteq B$.

Solution

(*a*) Here $(A - B) \cup (B - A) = (A \cap B') \cup (B \cap A')$ $[\because \ A - B = A \cap B]$

$= [(A \cap B') \cup B] \cap [(A \cap B') \cup A']$

(by distributive law)

$= [(A \cup B) \cap (B' \cup B)] \cap [(A \cup A') \cap (B' \cup A')]$

(by distributive law)

$= [(A \cup B) \cap U] \cap [U \cap (B' \cap A)]$ $[\because \ B' \cup B = U]$

$= (A \cup B) \cap [(B' \cap A')]$ $[\because \ A \cap U = A]$

$= (A \cup B) \cap (B \cup A)'$ (by De Morgan's law)

$= (A \cup B) - (B \cap A)$ $\because \ A \cap B = A - B'$ ■

(*b*) Here $A - B = \phi$

$\Rightarrow \quad A \cap B' = \phi$ $[\because \ A - B = A \cap B']$

$\Rightarrow \quad A, \ B'$ are disjoint

$\Rightarrow$ all elements of A belong to B

$\Rightarrow \quad A \subseteq B$

Conversely, $A \subseteq B$

$\Rightarrow$ there is no element in A which does not belong to B

$\Rightarrow \quad A - B = \phi$

Hence $A - B = \phi \quad \Leftrightarrow \quad A \subseteq B.$ ■

Example 27

Prove that

$$(A \cap B) \times (C \cap D) = (A \times C) \cap (B \times D)$$

Solution

Let $(x, y) \in (A \cap B) \times (C \cap D)$; then

$(x, y) \in (A \cap B) \times (C \cap D)$

$\Rightarrow$ $x \in A \cap B, y \in C \cap D$

$\Rightarrow$ $x \in A$ and $x \in B$, while $y \in C$ and $y \in D$

$\Rightarrow$ $x \in A, y \in C$ and $x \in B, y \in D$

$\Rightarrow$ $(x, y) \in A \times C$ and $(x, y) \in B \times D$

$\Rightarrow$ $(x, y) \in (A \times C) \cap (B \times D)$

$\therefore (A \cap B) \times (C \cap D) \subseteq (A \times C) \cap (B \times D)$ ...(*i*)

Conversely, $(a, b) \in (A \times C) \cap (B \times D)$

$\Rightarrow$ $(a, b) \in A \times C$ and $(a, b) \in B \times D$

$\Rightarrow$ $A \in A, b \in C$ and $a \in B, b \in D$

$\Rightarrow$ $(A \in A$ and $a \in B)$ and $(b \in C$ and $b \in D)$

$\Rightarrow$ $a \in A \cap B$ and $b \in C \cap D$

$\Rightarrow$ $(a, b) \in (A \cap B) \times (C \cap D)$

$\therefore (A \times C) \cap (B \times D) \subseteq (A \cap B) \times (C \cap D)$

Hence, by (*i*) and (*ii*), we get

$$(A \cap B) \times (C \cap D) = (A \times C) \cap (B \times D).$$ ■

Example 28

If the set of all positive integers be such that

$$(a, b) + (c, d) = (a + c, b + d)$$

and $$(a, b) \times (c, d) = (ac + bd, ad + bc)$$

prove that multiplication is distributive with respect to addition.

Solution

Let us have ordered pairs (a, b), (c, d), (e, f) with elements as positive integers. Here we are to prove

$$(a, b) \times [(c, d) + (e, f)] = (a, b) \times (c, d) + (a, b) \times (e, f)$$

Now, by given addition law for ordered pairs, we have

$$(c, d) + (e, f) = (c + e, d + f)$$

$$\therefore \quad (a, b) \times [(c, d) + (e, f)] = (a, b) \times (c + e, d + f)$$

$$= [a(c + e) + b(d + f), a(d + f) + b(c + e + d + f)]$$

(by product law)

$$= (ac + bd + ae + bf, af + be + ad + bc)$$

$$= [(ac + bd) + (ae + bf), (ad + bc) + (af + be)]$$

$$= (ac + bd, ad + bc) + (ae + bf, af + be)$$

(by addition law)

$$= (a, b) \times (c, d) + (a, b) \times (e, f)$$ by product law. ■

Example 29

If A and B are subsets of the universal set ∪, prove the following for any $x \in U$:

(*i*) $f_{A \cap B}(x) = f_A(x) f_B(x)$

(*ii*) $f_{A \cup B}(x) = f_A(x) + f_B(x) - f_A(x) f_B(x)$

(*iii*) $f_{A \oplus B}(x) = f_A(x) + f_B(x) - 2 f_A(x) f_B(x)$

Solution

(*i*) If $x \in A \cap B$, then $x \in A$ and $x \in B$, and we have

$$f_{A \cap B}(x) = 1, f_A(x) = 1, f_B(x) = 1$$

so that $$f_{A \cap B}(x) = 1 = f_A(x) f_B(x)$$

If $x \notin A \cap B$, then $x \in \overline{A \cap B} = \overline{A} \cup \overline{B}$; that is $x \in \overline{A}$ or $x \in \overline{B}$.

In this case, we have

$$f_{A \cap B}(x) = 0 \text{ and } f_A(x) = 0 \text{ or } f_B(x) = 0,$$

so that $$f_{A \cap B}(x) = 0 = f_A(x) f_B(x)$$

Thus, in both of the Cases : $x \in A \cap B$ and $x \notin A \cap B$, the given result is true.

(*ii*) If $x \in A \cup B$, then $x \in A$ or $x \in B$.

In this case $$f_{A \cup B}(x) = 1$$

and $$f_A(x) + f_B(x) - f_A(x) f_B(x) = \begin{cases} 1 + f_B(x) - f_B(x) = 1 & \text{when } x \in A \\ f_A(x) + 1 - f_A(x) = 1 & \text{when } x \in B \\ 1 + 1 - 1 = 1 & \text{when } x \in A \text{ and } x \in B \end{cases}$$

Hence, when $x \in A \cup B$, we have

$$f_{A\cup B}(x) = 1 = f_A(x) + f_B(x) - f_A(x) f_B(x)$$

If $x \notin A \cup B$, the $x \in \overline{A \cap B} = \overline{A} \cup \overline{B}$ so that x $\notin A$ and $x \notin B$.

In this case, $f_{A\cup B}(x) = 0, f_A(x) = 0, f_B(x) = 0$

so that $f_{A\cup B}(x) = 0 = f_A(x) + f_B(x) - f_A(x) f_B(x)$

Thus, in both of the cases : $x \in A \cup B$ and $x \notin A \cup B$, the given result is true.

(*iii*) Suppose $x \in A \oplus B = (A - B) \cup (B - A)$. Then $x \in A, x \notin B$ or $x \notin B$, $x \notin A$.

In this case, $f_{A\oplus B}(x) = 1$

and $f_A(x) + f_B(x) - 2f_A(x) f_B(x) = \begin{cases} 1 + 0 - 0 = 1 & \text{when } x \in A, x \notin B \\ 0 + 1 - 0 = 1 & \text{when } x \in B, x \notin A \end{cases}$

Hence, when $x \in A \oplus B$, we have

$$f_{A\oplus B}(x) = 1 = f_A(x) + f_B(x) - 2f_A(x) f_B(x)$$

Next, suppose $x \notin A \oplus B$. Then, either $x \notin \overline{A}$ and $x \in \overline{B}$, or $x \in A \cap B$.

In this case, $f_{A\oplus B}(x) = 0$

and $f_A(x) + f_B(x) - 2f_A(x) f_B(x) = \begin{cases} 0 + 0 - 0 = 0 & \text{if } x \in \overline{A} \text{ and } x \in \overline{B} \\ 1 + 1 - 2 = 0 & \text{if } x \in A \cap B \end{cases}$

Hence, when $x \notin A \oplus B$, we have

$$f_{A\oplus B}(x) = 0 = f_A(x) + f_B(x) - 2f_A(x) f_B(x).$$

Thus, in both of the cases : $x \in A \oplus B$ and $x \notin A \oplus B$, the given result is true. ■

Example 30

Find the elements of the set $A \times B \times C$, where

$$A = \{1, 2, 3\}, B = \{4, 5\}, C = \{6, 7, 8\}$$

Solution

Here the set $A \times B \times C$ will consist of $3 \times 2 \times 3 = 18$ ordered triads. *A* convenient method is the tree diagram given ahead:

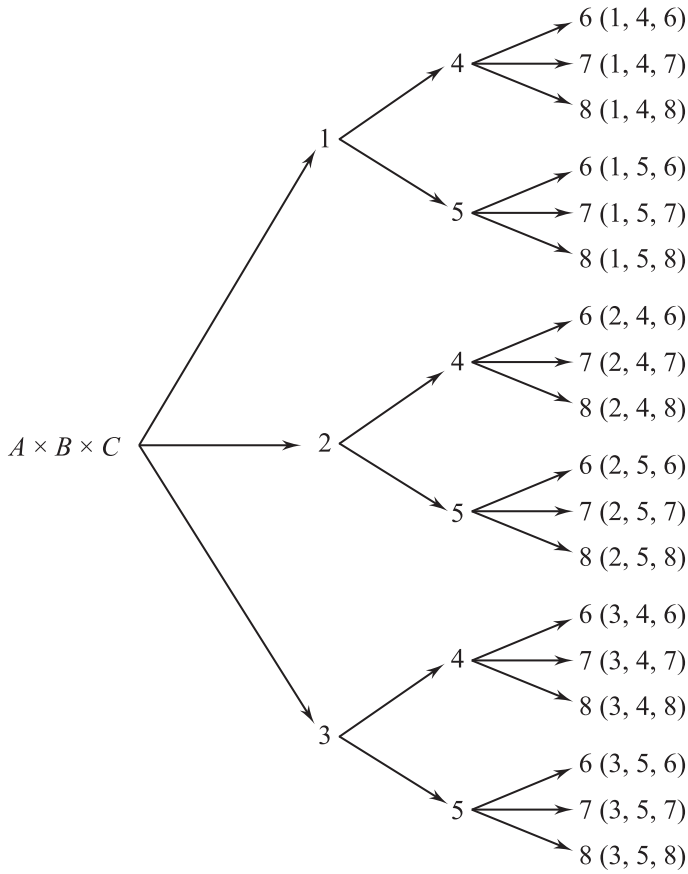


Fig. 1.8

Hence $A \times B \times C = \{(1, 4, 6), (1, 4, 7), (1, 4, 8), (1, 5, 6), (1, 5, 7)$
$(1, 5, 8), (2, 4, 6), (2, 4, 7), (2, 4, 8), (2, 5, 6), (2, 5, 7), (2, 5, 8),$
$(3, 4, 6), (3, 4, 7), (3, 4, 8), (3, 5, 6), (3, 5, 7), (3, 5,8)\}$.

■

Example 31

Using set laws, verify that

$$(X - Y) - Z = X - (Y \cup Z)$$

Solution

$$\begin{aligned}
(X - Y) - Z &= (X - Y) \cap Z' && (\because\ A - B = A \cap B') \\
&= (X \cap Y') \cap Z' && (A - B = A \cap B') \\
&= X \cap (Y' \cap Z') \\
&= X \cap (Y' \cap Z') && \text{(Associative law)} \\
&= X \cap (Y \cup Z)' && \text{(De Morgan's law)} \\
&= X - (Y \cup Z) && (A - B = A \cap B')
\end{aligned}$$

■

Example 32

Simplify the set expression

$$(A \cap B') \cup (A' \cap B) \cup (A' \cap B')$$

Solution

$$\begin{aligned}(A \cap B') \cup (A' \cap B) \cup (A' \cap B') &= (A \cap B') \cup [(A' \cap B) \cup (A' \cap B')] \\ &= (A \cap B') \cup [A' \cap (B \cup B')] \\ &= (A \cap B') \cup (A' \cap U) \\ &= (A \cap B') \cup A' \\ &= A' \cup (A \cap B') \\ &= (A' \cup A) \cap (A' \cup B') \\ &= U \cap (A' \cup B') \\ &= A' \cup B'.\end{aligned}$$

■

Example 33

If A, B and C are sets, prove that

$$A \cap (B - C) = (A \cap B) - (A \cap C)$$

Proof

$$\begin{aligned}A \cap (B - C) &= \{x : x \in A \text{ and } x \in (B - C)\} \\ &= \{x : x \in A \text{ and } (x \in B \text{ and } x \notin C)\} \\ &= \{x : x \in A \text{ and } (x \in B \text{ and } x \in C')\} \\ &= \{x : x \in (A \cap B \cap C')\} \\ &= A \cap B \cap C' \\ (A \cap B) - (A \cap C) &= \{x : x \in (A \cap B) \text{ and } x \in (A \cap C)'\} \\ &= \{x : x \in (A \cap B) \text{ and } x \in (A' \cup C')\} \\ &\qquad\qquad \text{(De Morgan's law)} \\ &= \{x : [x \in (A \cap B) \text{ and } x \in A'] \\ &\quad \text{or } [x \in (A \cap B) \text{ and } x \in C']\} \\ &= \{x : x \in (A \cap A' \cap B) \text{ or } x \in (A \cap B \cap C')\} \\ &= \{x : x \in \phi \quad \text{or} \quad x \in (A \cap B \cap C')\} \\ &= \{x : x \text{ e } A \cap B \cap C'\} \\ &= A \cap B \cap C'\end{aligned}$$

■

SUMMARY

Laws of Sets

Let A, B and C be any three sets and U the universal set, then.

Idempotent Laws

1. $A \cup A = A$ **2.** $A \cap A = A$

Identity Laws

3. $A \cup \phi = A$ **4.** $A \cap U = A$

Inverse Laws

5. $A \cup A' = U$ **6.** $A \cap A' = \phi$

Domination Laws

7. $A \cup U = U$ **8.** $A \cap \phi = \phi$

Commutative Laws

9. $A \cup B = B \cup A$ **10.** $A \cap B = B \cap A$

Double Complementation Law

11. $(A')' = A$

Associative Laws

12. $A \cup (B \cup C) = (A \cup B) \cup C$ **13.** $A \cap (B \cap C) = (A \cap B) \cap C$

Distributive Laws

14. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

15. $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

De-Morgan's Laws

16. $(A \cup B)' = A' \cap B'$ **17.** $(A \cap B)' = A' \cup B'$

Absorption Laws

18. $A \cup (A \cap B) = A$ **19.** $A \cap (A \cup B) = A$

Unnamed Laws

20. If $A \subseteq B$; then $A \cap B = A$ **21.** If $A \subseteq B$; then $A \cup B = B$

22. If $A \subseteq B$; then $B' \subseteq A'$ **23.** $A - B = A \cap B'$

24. $A \oplus B = A \cup B - A \cap B$.

EXERCISE

1. Prove that $(A - C) \cap (C - B) = \phi$ analytically, where A, B and C are sets. Verify graphically also.
2. If A, B and C are sets, prove both analytically and graphically, that $A - (B \cap C) = (A - B) \cup (A - C)$.
3. If A, B and C are sets, prove that $[A \cup (B \cap C)]' = (C' \cup B') \cap A'$, using set identities.
4. If A, B and C are sets, prove that $A \times (B \cap C) = (A \times B) \cap (A \times C)$.
5. If A, B, C and D are sets, prove that $(A \cap B) \times (C \cap D) = (A \times C) \cap (B \times D)$.
6. Using venn diagram to prove that $\oplus$ is an associative operation *viz.*, $(A \oplus B) \oplus C = A \oplus (B \oplus C)$.
7. Using venn diagram prove that $(A \oplus B) \times C = (A \times C) \oplus (B \times C)$ where A, B, C are sets.
8. Find the sets A and B if

   (*a*) $A - B = \{1, 3, 7, 11\}$, $B - A = \{2, 6, 8\}$ and $A \cap B = \{4, 9\}$

   (*b*) $A - B = \{1, 2, 4\}$, $B - A = \{7, 8\}$ and $A \cup B = \{1, 2, 4, 5, 7, 8, 9\}$

   [**Ans.** (*a*) $A = \{1, 3, 4, 7, 9, 11\}$, $B = \{2, 4, 6, 8, 9\}$

   (*b*) $A = \{1, 2, 4, 5, 9\}$, $B = \{5, 7, 8, 9\}$]
9. Prove that $(A \cap B) \cup (A \cap B \sim B) = A$ and $A \cap (\sim A \cup B) = A \cap B$.
10. Prove that $(A \cap B) \times (C \cap D) = (A \times C) \cap (B \times D)$.
11. Show that $(A \cap B) \cup C = A \cap (B \cup C)$ iff $C \subseteq A$.
12. Test whether the following is a valid argument:

    All men are mortal. Sachin is a man.

    Therefore, Sachin is mortal.
13. Test whether the following is a valid argument. All who study mathematics are intelligent. All students of college *XYZ* are intelligent. Therefore, some students of college *XYZ* study mathematics.
14. Prove that if $B \supset A$, $A \supset C$, $C \supset B$ then $A = B = C$.
15. Prove that a set consisting of n distinct elements yields $2^n - 1$ proper subsets.

Suggested Readings

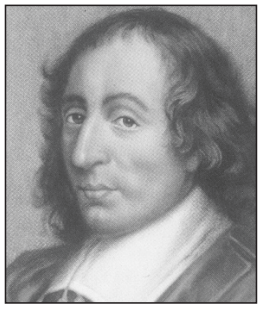
1. R.R. Chiristian, *Introduction to Logic and sets,* Blalsdell, Watham, MA, 1965.
2. P.R. Halmos, *Naive Set Theory,* Van Nostrand, New York, 1960.
3. Tramley and Monohar, *Discrete Mathematics for Sc & Engg.,* PHI. 1999.
4. S. Sahni, *Concepts in Discrete Mathematics,* 2nd ed; camelot, Fridley MN, 1985, pp. 93-111.
5. P.S. Suppes, *Aromatic Set theory,* Van Nostrand, New York, 1960.
6. R.L. Wilder, *Evolution of Mathematical Concepts: An Elementary study*, Wiley, New York, 1968.
7. S.K. Yadav, *Discrete Mathematical structures,* Satya Prakashan, New Delhi, 2001.

❑❑❑

Check for updates

2

Basic Combinatorics



Blaise Pascal (1623–1662)

Blaise Pascal (1623–1662) was born in Clermont-Ferand, France. He showed phenomenal mathematical ability at early age. He was encouraged by his father to pursue subjects such as ancient languages. His father refused him to teach any science until he found that Pascal discovered many theorems in elementary geometry at the age of 12. At 14 he started attending the weekly meetings of a group of French mathematicians that later became the French Academy. At 16 he successfully developed important results in conic section on which he wrote a book. At 19 he invented the first mechanical calculating machine.

In 1650 he felt health problem and left the mathematical and scientific work for three years and pursued religion. He worked with fermat and laid the foundation of probability theory. The programming language Pascal is named after him. Most of his life was spent in physical pain and poor health. He left this world in early age of 39.

2.1 Introduction

An important part of discrete mathematics is combinatorics which is used to solve counting problems without enumerating all possible cases. Combinatorics deals with counting the number of ways of arranging or choosing objects from a finite set according to certain specific rules or methods. Frankly speaking, combinatorics is concerned with problems of permutations and combinations, which we are familear with. Combinatorics has wide applications in computer science, especially in coding theory and cryptography, analysis of algorithms and probability theory.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_2

According to Prof. H.L. Royden, a set is called countable if it is the range of some sequence and finite if it is the range of some finite sequence. A set that is not finite is termed as infinite (Many authors restrict the use of the word 'countable' to sets that are finite and countable, but we include the finite sets among the countable sets). We use the term 'countably infinite' for infinite countable sets.

A set is called finite if it is either empty or the range of a finite sequence. A set is called countable (or denumerable) if it is either empty or the range of a sequence.

2.2 Basic Counting Principles

We consider an event to occur in n_1 different ways, and if, to follow this, a second event occurs in n_2 different ways and following this second event a third event occurs in n_3 different ways and so on. Then the number of ways the events can occur in the order indicated is $n_1, n_2, n_3, ...$

The following rules have to be considered in calculations.

2.2.1 The Principle of Disjunctive Counting (Sum Rule)

Let a set X be the union of disjoint non-empty subset $S_1, S_2, ..., S_n$.

Then $|X| = |S_1| + |S_2| + |S_3| + ... + |S_n|$

We can emphasize that the subsets $S_1, S_2, S_3, ..., S_n$ must have no elements in common.

Since $X = S_1 \cup S_2 \cup S_3 \cup ... \cup S_n$, each element of X is in exactly once of the subsets S_i.

In other words $S_1, S_2, S_3, ..., S_n$ is a partition of X.

If the subsets $S_1, S_2, S_3, ..., S_n$ are allowed to overlap then a more profound principle will be needed which is called the principle of inclusion and exclusion.

Let $E_1, E_2, ..., E_n$ be mutually exclusive events and E_1 can happen in e_1 ways, E_2 can happen e_2 ways. Then E_1 or E_2 or ... or E_n can happen in $e_1 + e_2 + ... + e_n$ ways.

This rule can be formulated in terms of choices. If an object can be selected from a reservois in e_1 ways and an object can be selected from a separated reservoir in e_2 ways, then the selected one object from either one reservoir or the other can be made in $e_1 + e_2$ ways.

2.2.2 The Principle of Sequential Counting (Product Rule)

Let $S_1, S_2, ..., S_n$ be non-empty sets, then the number of elements is a cartesian product.

$S_1 \times S_2 \times S_3 \times ... \times S_n$ is the product denoted by $\prod_{i=1}^{n} |S_i|$

i.e., $|S_1 \times S_2 \times S_3 \times ... \times S_n| = \prod_{i=1}^{n} |S_i|$

Illustration

Let $S_1 \times S_2$ be a tree where

$$S_1 = \{a_1, a_2, a_3, a_4, a_5\} \text{ and } S_2 = \{b_1, b_2, b_3\}$$

We can observe that there are 5 branches in the first stage corresponding to the 5 elements of S_1 and to each of these branches there are three branches is the second stage corresponding to the 3 elements of S_2 giving a total of $5 \times 3 = 15$ branches altogether.

The cartesian product $S_1 \times S_2$ can be partitioned as

$$(a_1 \times S_2) \cup (a_2 \times S_2) \cup (a_3 \times S_2) \cup (a_4 \times S_2) \cup (a_5 \times S_2)$$

where $(a_i \times S_2) = \{(a_i, b_1), (a_i, b_2), (a_i, b_3)\}$

i.e., $(a_3 \times S_2)$ corresponds to the third branch in the first stage followed by each of the 3 branches in the second stage.

If $a_1, a_2, ..., a_n$ be the n distinct elements of S_1 and $b_1, b_2, ..., b_m$ be the m distinct elements of S_2, then

$$S_1 \times S_2 = \bigcup_{i=1}^{n} (a_i \times S_2)$$

If x be an arbitrary element of $S_1 \times S_2$ then

$$x = (a, b) \text{ where } a \in S_1 \text{ and } b \in S_2.$$

Thus, $a = a_i$ for some i and $b = b_j$ for some j

Then, $x = (a_i, b_j) \in (a_i \times S_2)$

Therefore, $x \in \bigcup_{i=1}^{n} (a_i \times S_2)$

Conversely,

If $x \in \bigcup_{i=1}^{n} (a_i \times S_2)$

Then $x \in (a_i \times S_2)$ for some i and

$x = (a_i, b_j)$ where b_j is some element of S_2.

Therefore, $x \in S_1 \times S_2$.

We can observe that,

$(a_i \times S_2)$ and $(a_j \times S_2)$ are disjoint if $i \neq j$

If $x \in (a_i \times S_2) \cap (a_j \times S_2)$

Then $x = (a_i, b_k)$ for some k

and $x = (a_j, b_l)$ for some l

But then $(a_i, b_k) = (a_j, b_l)$

$\Rightarrow$ $a_j = a_i$ and $b_k = b_l$

But since $i \neq j, a_i \neq a_j$

Thus, we can conclude that,

$S_1 \times S_2$ is the disjoint union of the sets $(a_i \times S_2)$

Furthermore, $| a_i \times S_2 | = | S_2 |$ (One-to-one correspondence is $a_i \times S_2$ and S_2)

Namely, $(a_i, b_j) \Rightarrow b_j$

by the sum rule

$$| S_1 \times S_2 | = \sum_{i=1}^{n} | a_i \times S_2 | = (n \text{ summands})$$

$$| S_2 | + | S_2 | + | S_2 | + \ldots + | S_2 | = n | S_2 | = nm.$$

Note: (*i*) We can reformulate the product rule is events.
If $E_1, E_2, E_3, \ldots, E_n$ be heapend in $e_1, e_2, e_3, \ldots, e_n$ ways respectively, then the sequences of events E_1 first, followed by E_2, ... , followed by E_n can be happen in $e_1, e_2, e_3, \ldots, e_n$ ways.

(*ii*) In terms of choices, the product rule is stated thus. If a first object can be chosen in e_1 ways, a second is e_2 ways, ..., and an nth object can be chosen in en way then the choice of a first, second, and nth object can be made in $e_1, e_2, \ldots, e_n$ ways.

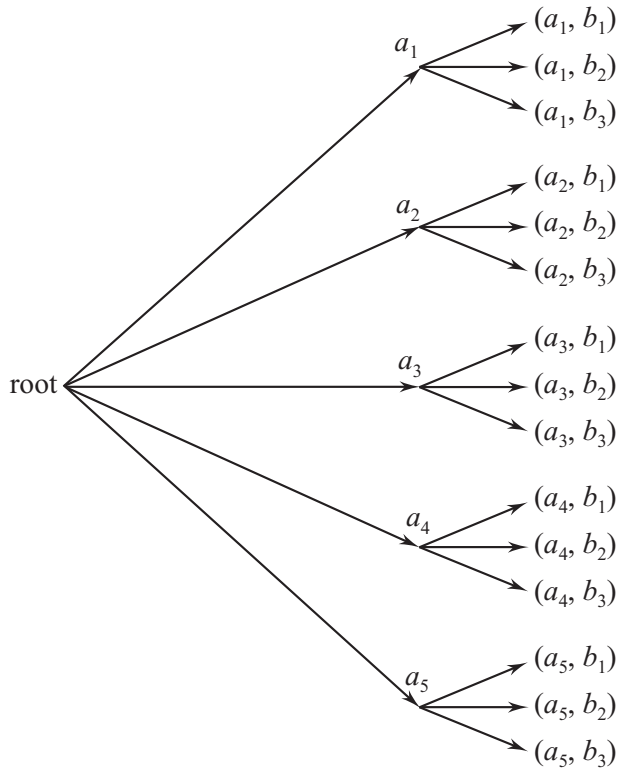


Fig. 2.1: Tree diagram of product rule.

2.3 Factorial

Each positive integer is defined as factorial as the product of all integers from 1 to n as

$$n\,! = n(n-1)(n-2)\ldots 3.2.1$$

(Product of integers from 1 to n)

We define $0\,! = 1$ and $1\,! = 1$

Hence $5\,! = 5.4.3.2.1 = 120$

and $6\,! = 6.5.4.3.2.1 = 720$

Factorial n is written as $n\,!$ or $\underline{|n}$

from above we can conclude $n\,! = n[(n-1)\,!]$.

Example 1

How many 3*-digit numbers can be formed using the digits* 1, 3, 4, 5, 6, 8 *and* 9*? How many can be formed if no digit can be repeated?*

Solution

There are 73 such 3-digit numbers in. Since each of 3 digits can be filled with 7 possibilities likewise, the anwer to question. In second part 7 : 6 : 5 there are 7 possibilities for the hundred's digit but once one digit is used it is not available for the ten's digit. There are only 6 possibilities for the ten's digit, and then for same reason there are only 5 possibilities for the unit digit. ■

Example 2

How many different license plates are there that involve 1, 2 *or* 3 *letter followed by* 1, 2, 3, *or* 4 *digits?*

Solution

We can see that there are $(26 + 26^2 + 26^3)10$ ways to form plates of 1, 2, or 3 letters followed by 1 digit, $(26 + 26^2 + 26^3)10^2$ plates of 1, 2, or 3 letters followed by 2 digits. $(26 + 26^2 + 26^3)10^3$ plates of 1, 2, or 3 letters followed by 3 digits and $(26 + 26^2 + 26^3)10^4$ plates of 1, 2, or 3 letters followed by 4 digits. We can apply the sum rule to conclude that there are $(26 + 26^2 + 26^3)10 + (26 + 26^2 + 26^3)10^2 + (26 + 26^2 + 26^3)10^3 + (26 + 26^2 + 26^3)10^4 = (26 + 26^2 + 26^3)\ (10 + 10^2 + 10^3 + 10^4)$ ways to form plates of 1, 2, or 3 letters by 1, 2, 3 or 4 digits.

Example 3

How many three-digit numbers are there which are even and have no repeated digits? (Here we are using all digits 0 *through* 9*).*

Solution

For a number to be even it must end in 0, 2, 4, 6 or 8. There are two cases to consider. First, suppose that the number ends in 0; then there are 9 possibilities for the first digit and 8 possibilities for the second since no digit can be repeated. Hence there are 9 . 8 three-digit numbers that end in 0. Now suppose the number does not end in 0. The there are 4 choices for the last digit (2, 4, 6 or 8); when this digit is specified, then there are only 8 possibilities for the first digit, since the number cannot begin with 0. Finally, there are 8 choices for the second digit and therefore there are 8 . 8 . 4 numbers that do not end in 0. Accordingly since these two cases are mutually exclusive, the sum rule gives 9 . 8 + . 8 . 4 even three-digit numbers with no repeated digits. ■

Example 4

A telegraph can transmit two different signals : a dot and a dash. What length of these symbols is needed to encode 26 *letters of the English alphabet and the ten digits* 0, 1, 2, ... 9*?*

Solution

Since there are two choices for each character, the number of different sequences of length k is 2^k. Therefore, the number of non-trivial sequences of length n or less is

$$2 + 2^2 + 2^3 + ... + 2^n = 2^{n+1} - 2$$

To encode 26 letters and 10 digits, we require at least 36 sequences of the above type; that is

$$2^{n+1} - 2 \geq 36$$

The least value of n (positive integer) for which this inequality holds is $n = 5$.

Hence, the length of the symbols needed to encode 26 letters and 10 alphabets is at least 5. ■

Example 5

A six faced die is tossed four times and the numbers shown are recorded in a sequence. How many different sequences are there?

Solution

In each toss, there are six possible numbers that the die can show. That is, for each place in the sequence, there are 6 possible entries. Therefore, the total number of possible sequences (each having 4 elements) is

$$6 \times 6 \times 6 \times 6 = 64 = 1296.$$ ■

2.4 Permutation and Combination

Ordered arrangement of r elements of a set containing n distinct elements is called in r-permutation of n elements and denoted by $P(n, r)$ or ${}^nP_r$ where $n \geq r$.

An unordered selection of r elements of a set containing n distinct elements is called an r combination of n elements and denoted by $C(n, r)$ or ${}^nC_r$ or $\binom{n}{r}$.

C.T.M: A permutation of objects involves ordering while a combination does not take ordering into account.

The first element of the permutation can be selected from a set having n elements in n ways. Having selected the first element for the first position of the permutation, the second element can be selected in $(n-1)$ ways, as there are $(n-1)$ elements left in the set.

Similarly, there are $(n-2)$ ways to select the consecutive third element and so on. There are $n-(r-1) = n-r+1$ ways to select the rth element.

By product rule, there are

$$n(n-1)\,(n-2)\,...(n-r+1)$$

Way for ordered arrangements of r elements of the given set.

Thus, $$P(n, r) = n(n-1)\,(n-2)\,(n-3)\,...\,(n-r+1)$$

$$= \frac{n!}{(n-r)!}$$

Particularly $P(n, n) = n!$

In r successive steps and step 1 can be done in n_1 ways, step 2 can be done in n_2 ways, ..., step r can be done in n_r ways, then the activity can be done in $(n_1, n_2, ..., n_r)$ ways.

The r permutations of the set can be obtained by first forming the $C(n, r)$, r-combinations of the set and then ordering the elements in each r-combinations, which can be done in $P(r, r)$ ways. Thus,

$$P(n, r) = C(n, r),\, P(r, r)$$

$$\Rightarrow \quad C(n, r) = \frac{P(n, r)}{P(r, r)} = \frac{n!/(n-r)!}{r!/(r-r)!}$$

$$= \frac{n!}{r!\,(n-r)!}$$

Particularly, $C(n, n) = 1$

Note: (*i*) This is called product rue.

(*ii*) Since the number of ways of selecting out r elements from a set of n elements is the same as the number of ways of leaving $(n-r)$ elements in the set

then $C(n, r) = C(n, n-r)$

(This is called complementary combination)

Important Theorems on Permutations and Combinations

Theorem 1

The number of distinguishable permutations of n objects in which the first object appears K_1 times, the second object appears K_2 times and so on, is

$$\frac{n!}{K_1!\,K_2!\ldots K_r!}$$

where K_r, the rth object, appears K_r times.

Proof

Let x be the number of permutations of n objects taken all at a time.

i.e., $$x = P(n, n) = n!$$

In permutation among x, make K_1 all distinct. Then since K_1 objects can be permuted among themselves, one permutation will give rise to $K_1!$ permutations.

$\therefore$ x permutations give rise to $x.\ K_1!$ permutations. Now make K_2 identical objects all distinct. Then we have $xK_1!\,K_2!$ permutations of n objects in which K_3 are alike ... K_r are alike. Continuing this process we get $x\ .\ K_1!\ .\ K_2!\ \ldots\ K_r!$ as the number of permutations of n objects of which are all distinct and hence equal to $n!$.

$\therefore$ $x\ .\ K_1!\,K_2!\ldots K_r! = n!$

i.e., $$x = \frac{n!}{K_1!\,K_2!\ldots K_r!}$$ ■

Theorem 2

Let A be a set of n elements and $1 \le r \le n$. Then the number of combinations of n elements of A, taken r at a time is $\dfrac{n!}{r!\,(n-r)!}$ *and it is denoted by C(n, r)*

i.e., $$C(n, r) = \frac{n!}{r!\,(n-r)!}$$

Theorem 3

Suppose that r selections are to be made from n items without regard to order and that repeats are allowed, assuming at least r copies of the n items. The number of ways these selections can be made is

$$C(n + r - 1, r) = \frac{(n + r - 1)!}{r!\,(n - 1)!} \qquad \left[\because \quad C(n, r) = \frac{n!}{r!\,(n - r)!}\right]$$

2.4.1 Cyclic Permutation

If the objects are arranged in a circle (or any closed curve) we get circular permutation and the number of circular permutations will be different from the number of linear permutations as the following example.

If we are to arrange 4 elements A, B, C, D is a circle. We have to fix one of the elements (say A) at the top point of the circle. The other 3 elements (B, C, D) are permuted is all possible ways, resulting in 3 ! = 6 different circular permutations as shown in the figure below:

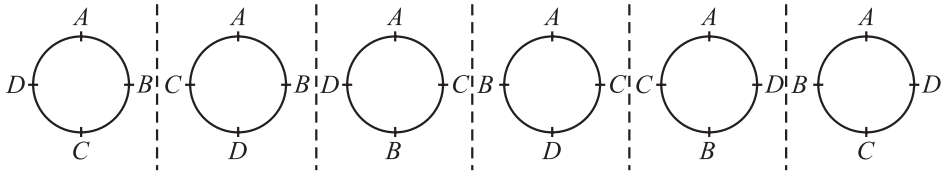


Fig. 2.2: Cyclic Permutation

Similarly, the number of different circular arrangements of n objects = $(n - 1)$! If no distinction is made between clockwise and counterclockwise circular arrangements (For example, if the circular arrangements in the first and the last figures are assumed as the same), then the number of different circular arrangements = $\frac{1}{2}(n - 1)$!

2.4.2 Pascal's Identity

This is a practical problem as follows:

If n and r be positive integers, where n d r

Then $C(n, r - 1) + C(n, r) = C(n + 1, r)$

This can be proved as:

Let S be a set containing $(n + 1)$ elements, one of which is 'a'.

Let $S' \equiv S - \{a\}$

The number of subsets of S containing r element is $C(n + 1, r)$

Now, a subset of S with r elements either contains 'a' together with $(r - 1)$ elements of S' or contains r elements of S' which do not include 'a'.

The number of subsets of $(r - 1)$ elements of $S' = C(n, r - 1)$

$\therefore$ The number of subsets of r elements of S that contains 'a' $= C(n, r - 1)$

Also the number of subsets of r elements of S that do not contain 'a' = that of $S' = C(n, r)$

Consequently, $C(n, r - 1) + C(n, r) = C(n + 1, r)$.

Corollary

$$C(n + 1, r + 1) = \sum_{i=r}^{n} C(i, r)$$

Proof

Changing n to i and r to $r + 1$ in Pascal's identity, we get

$$C(i, r) + C(i, r + 1) = C(i + 1, r + 1)$$

i.e.,
$$C(i, r) = C(i + 1, r + 1) - C(i, r + 1) \qquad ...(i)$$

Putting $i = r, r + 1, ..., n$ in (i) and adding, we get

$$\sum_{i=r}^{n} C(i, r) = C(n + 1, r + 1) - C(r, r + 1)$$
$$= C(n + 1, r + 1) \qquad [\because \; C(r, r + 1) = 0]$$

2.4.3 Vandermonde's Identity

Let m, n and r be non-negative integers, where $r \leq m$ or n

Than $$C(m + n, r) = \sum_{i=0}^{r} C(m, r - i) \; . \; C(n, i)$$

To prove this identity,

We consider m and n as the number of elements is sets 1 and 2 respectively. Then the total number of ways of selecting r element from the union of sets 1 and 2 = $C(m + n, r)$.

The r elements can also be selected by selecting r elements from set 2 and $(n - i)$ elements from set 1, where $i = 0, 1, 2, ..., r$.

This can be selected as $C(m, r-i)\ .\ C(n, i)$ ways.

Using product rule

The $(r+1)$ selections corresponding to $i = 0, 1, 2, ..., r$ are disjoint.

Hence by sum rule

$$C(m+n, r) = \sum_{i=0}^{r} C(m, r-i)\ .\ C(n, i)$$

or $$\sum_{i=0}^{r} C(m, i)\ .\ C(n, r-i)$$

2.4.4 Pigeonhole Principle

It is a well known technique in mathematics used for counting principle. It is also known as shoe box argument or dirichlet drawer principle.

Statement

If 'm' pigeons are assigned to 'n' pigeonholes and $m > n$, then at least one pigeonhole contains two or more pigeons.

Proof

Assign one pigeon to each of the 'n' pigeon holes. Since $m > n$, there are still $(m-n)$ pigeons to be assigned to a pigeonhole.

Therefore, at least one pigeonhole will be assigned to a second pigeon.

Theorem

If n pigeons are assigned to m pigeonholes, then one of the pigeonholes will contain at least $\left(\frac{n-1}{m}+1\right)$ *pigeons.*

Proof

We prove the theorem by the method of contradiction. Assume that every pigeonhole will contain no more than $\left(\frac{n-1}{m}\right)$ pigeons. Then the total number of pigeons in the m pigeonholes will be at most equal to $m\left(\frac{n-1}{m}\right) = n-1$. This is a contradiction (because the number of pigeons is equal to n). Hence, our assumption is wrong.

This means that at least one pigeonhole will contain more than $\left(\frac{n-1}{m}\right)$ pigeons. This is equivalent to saying that (at least) one pigeonhole will contain at least $\left(\frac{n-1}{m}+1\right)$ pigeons. ■

2.4.5 Inclusion–Exclusion Principle

This principle is a counter part of arrangements of counting principles and set theory. As we have done is set theory.

If A and is be finite subsets of a finite universal set U, then

$$|A \cup B| = |A| + |B| - |A \cap B|$$

where $|A|$ denotes the cardinality of the set A.

This principle can be extended to a finite number of finite sets $A_1, A_2, ..., A_n$ as follows:

$$|A_1 \cup A_2 \cup ... \cup A_n| = \sum_i |A_i| - \sum_{i<j} |A_i \cap A_j| + \sum_{i<j<k} |A_i \cap A_j \cap A_k| - ... + (-1)^{n+1} |A_1 \cap A_2 \cap ... \cap A_n|,$$

where the first sum is over all i, the second sum is over all pairs i, j with $i < j$, the third sum is over all triples i, j, k with $i < j < k$ and so on.

Proof

Let $A \backslash B = \{a_1, a_2, ..., a_r\}$

$B \backslash A = \{b_1, b_2, ..., b_s\}$

$A \cap B = \{x_1, x_2, ..., x_t\}$,

where $A \backslash B$ is the set of those elements A which are not in B.

Then $A = \{a_1, a_2, ..., a_r, x_1, x_2, ..., x_t\}$

and $B = \{b_1, b_2, ..., b_s, x_1, x_2, ..., x_t\}$

Hence, $A \cup B = \{a_1, a_2, ..., a_r, x_1, x_2, ..., x_t, b_1, b_2, ..., b_s\}$

Now $|A| + |B| - |A \cap B| = (r + t) + (s + t) - t$

$= r + s + t = |A \cup B|$ ...(1)

Let us now extend the result to 3 finite sets A, B, C.

$|A \cup B \cup C| = |A \cup (B \cup C)|$

$= |A| + |B \cup C| - |A \cap (B \cup C)|$

$= |A| + |B| + |C| - |B \cap C| - \{(A \cap B) \cap (A \cap C)\}$ by (1)

$$= |A| + |B| + |C| - |B \cap C| - \{|A \cap B| + |A \cap C| - |(A \cap B) \cap (A \cap C)|\}, \text{ by (1)}$$

$$= |A| + |B| + |C| - |A \cap B| - (B \cap C) - (C \cap A) + |A \cap B \cap C|$$

Example 6

An identifier in a programming language consists of letter followed by alphanumeric characters. Find the number of legal identifiers of length atmost 10.

Solution

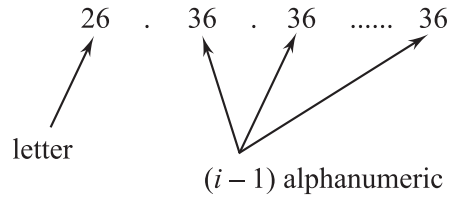
Let S_i denotes the set of identifiers of length i,

Where $1 \le i \le 10$

Then $|S_i| = 26 \,.\, 36^{i-1}$

Since the subtasks S_1, ..., S_{10} are mutually exclusive, by the addition principle.

The total number of identifiers of length ≤ 10 is given by



$$\sum_{i=1}^{10} |S_i| = \sum_{i=1}^{10} 26 \,.\, 36^{i-1} = 26\left(\sum_{i=0}^{9} 36^i\right)$$

$$= 26 \,.\, \left(\frac{36^{10} - 1}{36 - 1}\right) \text{ [G.P.]}$$

$$= \frac{26(36^{10} - 1)}{36 - 1} = \frac{26(36^{10} - 1)}{35}$$

$$= 2, 716, 003, 412, 618, 210$$

$$\approx 2.7 \text{ quadrillion.}$$ ■

Example 7

An eight-bit word is called a byte. Find the number of bytes with their second bit 0 or the third bit 1.

Solution

Number of bytes with second bit 0 = 2.1 . 2.2 . 2.2 . 2.2 = 2^7

Number of bytes with third bit 1 = 2 . 2.1 . 2.2 . 2 . 2.2 = 2^7

Since these two subtasks are not mutually exclusive, we cannot add these two partial answers and claim that the answer is

$$2^7 + 2^7 = 128 + 128 = 256.$$

So, we must find the number of bytes that have both properties. The number of bytes with second bit 0 and third bit 1 equals

$$2.1 . 1 . 2.2 . 2 . 2.2 = 2^6 = 64$$

So, by the inclusion–exclusion principle, the number of bytes with the given properties is

$$2^7 + 2^7 – 2^6 = 128 + 128 – 64 = 192.$$ ■

Example 8

The number of r permutations of n distinct elements satisfies the recurrence relation

$$P(n, r) = P(n – 1, r) + rP(n – 1, r – 1), \text{ where } 0 < r < n$$

Solution

Let X be a set with n elements and x and arbitrary element in it. The set of r-permutations of X can be partitioned into two subsets: A, the set of permutations not containing x, and B, the set of permutations containing x.

Two operations take place:

(*i*) To find the number of elements in *A*: Since no permutations in A contain x, every element in A is an r-permutation of $n – 1$ elements. The number of such permutations is $P(n – 1, r)$.

(*ii*) To find the number of elements in *B*: Since every permutations in B contains x, $n – 1$ candidates are left in X for the remaining $r – 1$ positions. They can be arranged in $P(n – 1, r – 1)$ ways. Now the position of x is a permutations has r choices. Therefore, by the multiplication principle $rP(n – 1, r – 1)$ permutations contain x.

Since A and B are disjoint sets, hence by addition principle

$$P(n, r) = |A| + |B|$$
$$= P(n-1, r) + rP(n-1, r-1).$$ ■

Example 9

(The Pizza problem) *Let f_n denotes the maximum number of places into which a pizza can be divided with n cuts. Find a formula for f_n.*

Solution

The maximum number of regions can be realized when every two chords, that is, cuts, intersect and no three chords are concurrent.

f_n can be defined recursively as

$$f_0 = 1$$
$$f_n = f_{n-1} + n, \quad n \geq 1$$

Solving relation,

$$f_n = 1 + \frac{n(n+1)}{2}$$

Thus, can be rewritten as,

$$f_n = 1 + n + \frac{n(n+1)}{2}$$
$$= C(n, 0) + C(n, 1) + C(n, 2), \quad n \geq 0$$ ■

Example 10

Let A be a 10-element subset of the set $\{1, 2, ..., 15\}$. Let A_s be a subset of A containing three elements, where the subscripts denotes the sum of the elements in A_s. For example, the subset $\{2, 4, 5\}$ is denoted by A_{11}. Determine if each subset of A can be identified by a unique name A_s. In other words, does ever sum $i + j + k$ have a unique value s, where $1 \leq i < j < k \leq 15$?

Solution

We let the pigeonhole principle do the job so far for us. The least value of s is $1 + 2 + 3 = 6$ and the largest value of s is $13 + 14 + 15 = 42$

Thus, $6 \leq s \leq 42$

There are at most 37 possible values of s.

There are $C(10, 3) = 120$

Three-element subsets (pigeons) of A and only 37 possible sums (pigeonholes), so, by the pigeonhole principle atleast two subsets must yield the same sum, that is not every three-element subset of A can have a unique name.

e.g., Let $A = \{1, 2, ..., 10\}$

Since subsets $\{1, 2, 5\}$ and $\{1, 3, 4\}$ yield the same sum, 8, they have the same name, A_8. ■

Example 11

Let $a_1, a_2, a_3, ..., a_n$ *represent an arbitrary arrangement of the numbers* 1, 2, 3, ..., n. *Prove that, if n is odd, then the product*

$$(a_1 - 1)\,(a_2 - 2)\,(a_3 - 3)\,...\,(a_n - n)$$

is an even number.

Solution

Since n is odd, we may write $n = 2k + 1$, where k is a positive integer, and the given product has $n = 2k + 1$ factors. Moreover, of the numbers 1, 2, 3, ..., $n = 2k + 1$, exactly $k + 1$ are odd. Since $a_1, a_2, ... a_n$ is just an arrangement (permutation of the numbers 1, 2, 3, ... n (= $2k + 1$), exactly $k + 1$ of the a's are odd. Therefore, among the numbers $a_1, a_2, ... a_n$; 1, 2, 3, ... n appearing in the given product, exactly $(k + 1) + (k + 1) = n + 1$ are odd. But there are only n factors in the product. Hence, (by pigeonhole principle) at least one of the factors contains two odd numbers, say a_m and m so that $a_m - m$ is even. Thus, at least one of the factors in the given product is even (divisible by 2). Therefore, the entire product is even. ■

Example 12

Suppose that a patient is given a prescription of 45 *pills with the instructions to take at least one pill per day for* 30 *days. Prove that there must be a period of consecutive days during which the patient takes a total of exactly* 14 *pills.*

Solution

Let a_i be the number of pills the patient has taken through the end of the ith day. Since the patient takes at least one pill per day at most 45 pills in 30 days, we have

$$1 \le a_1 < a_2 < a_3 < ... < a_{30} \le 45.$$

These inequalities give

$$a_1 + 14 < a_2 + 14 < a_3 + 14 < ... < a_{30} + 14 \leq 45 + 14 \ (= 59)$$

We now have 60 positive integers

$$a_1, a_2, a_3, ... a_{30}, ... a_{30}, a_1 + 14, a_2 + 14, ..., a_{30} + 14$$

all of which lie between 1 and 59 (both inclusive). (That is, we have 60 pigeons in 59 pigeon holes). So two of these numbers must be equal. Since $a_1, a_2, ... a_{30}$ are all different and $a_1 + 14, a_2 + 14, ... a_{30} + 14$ are all different, it must be that one of $a_1, a_2, ... a_{30}$ is equal to of $a_1 + 14, a_2 + 14, ... a_{30} + 14$. This means that there are i and j such that

$$a_i = a_j + 14$$

Thus, between days i and j, the patient takes exactly 14 pills. ■

Example 13

Show that among 100000 *people, there are* 2 *born at the same time (hour, minute, seconds).*

Solution

Let $P = \{P_1, P_2, ... P_{100000}\}$ be that set of people.

Let $H = \{H_1, H_2, ... H_{24}\}$ be the set of all hours in a day.

A function 'f' exists such that $f : P \to H$.

By extended pegionhole principle, there are at least $\left\lceil \frac{|P|}{|H|} \right\rceil$ persons born during the same hour $= \left\lceil \frac{100000}{24} \right\rceil = 4167$.

Let $Q = \lceil q_1, q_2, ... q_{4167} \rceil$ be the set of all persons born during the same hour.

Let $M = \{m_1, m_2, ... m_{60}\}$ be the set of minutes in an hour. Then T; $Q \Rightarrow M$. By extended pigonhole principle, the persons born during the same hour and same minute

$$= \left\lceil \frac{|Q|}{|m|} \right\rceil = \left\lceil \frac{4167}{60} \right\rceil = 70$$

Let $R = \{R_1, R_2, ..., R_{70}\}$ be the set of all persons born on same hour and same minute and $S = \{S_1, S_2, ... S_{60}\}$ be the set of all seconds in a minute.

Then $h : R \to S$.

By extended pigonhole principle, the number of persons born in same hour, minute and seconds = $\left\lceil \frac{|R|}{|S|} \right\rceil = \left| \frac{70}{60} \right| = 2.$ ■

Example 14

Find the least number of ways of choosing three different numbers from 1 *to* 10 *so that all choices have the same sum.*

Solution

From the numbers from 1 to 10, we can choose three different numbers in ${}^{10}C_3 = 120$ ways.

The smallest possible sum that we get from a choice is $1 + 2 + 3 = 6$ and the largest sum is $8 + 9 + 10 = 27$. Thus, the sums vary from 6 to 27 (both inclusive), and these sums are 22 in number.

Accordingly, here, there are 120 choices (pigeons) and 22 sums (pigeonholes). Therefore, the least number of choices assigned to the same sum is, by the generalized pigeonhole principle,

$$\left(\frac{120-1}{22}\right) + 1 = 6.4 \approx 6$$ ■

Example 15

(*a*) *Assuming that repetitions are not permitted, how many four-digit numbers can be formed form the six digits 1, 2, 3, 5, 7, 8?*

(*b*) *How many of these numbers are less than 4000?*

(*c*) *How many of the numbers in part (a) are even?*

(*d*) *How many of the numbers in part (a) are odd?*

(*e*) *How many of the numbers is part (a) are multiples of 5?*

(*f*) *How many of the numbers in part (a) contain both the digits 3 and 5?*

Solution

(*a*) The 4-digit number can be considered to be formed by filling up 4 blank spaces with teh available 6 digits. Hence, the number of 4-digits numbers

= the number of 4-permutations of 6 numbers

$= P(6, 4) = 6 \times 5 \times 4 \times 3 = 360$

(*b*) If a 4-digit number is to be less than 4000, the first digit must be 1, 2, or 3. Hence the first space can be filled up in 3 ways. Corresponding to any one of these 3 ways, the remaining 3 spaces can be filled up with the remaining 5 digits in $P(5, 3)$ ways. Hence, the required number = $3 \times P(5, 3)$

$$= 3 \times 5 \times 4 \times 3 = 180.$$

(*c*) If the 4-digit number is to be even, the last digit must be 2 or 8. Hence, the last space can be filled up in 2 ways. Corresponding to any one of these 2 ways, the remaining 3 spaces can be filled up with the remaining 5 digits in $P(5, 3)$ ways. Hence the required number of even numbers

$$= 2 \times P(5, 3) = 120.$$

(*d*) Similarly the required number of odd numbers = $4 \times P(5, 3) = 240$.

(*e*) If the 4-digit number is to be a multiple of 5, the last digit must be 5. Hence, the last space can be filled up in only one way. The remaining 3 spaces can be filled up in $P(5, 3)$ ways.

Hence, the required number = $1 \times P(5, 3) = 60$.

(*f*) The digits 3 and 5 can occupy any 2 of the 4 places in $P(4, 2) = 12$ ways.

The remaining 2 places can be filled up with the remaining 4 digits in $P(4, 2) = 12$ ways. Hence, the required number = $12 \times 12 = 144$. ■

Example 16

How many positive integers n can be formed using the digits 3, 4, 4, 5, 5, 6, 7, *if n has to exceed* 50,00,000?

Solution

In order that n may be greater than 50,00,000, the first place must be occupied by 5, 6 or 7.

When 5 occupies the first place, the remaining 6 places are to be occupied by the digits 3, 4, 4, 5, 6, 7.

The number of such numbers

$$= \frac{6!}{2!} \qquad [\because \text{ The digit 4 occurs twice}]$$

$$= 360$$

When 6 (or 7) occupies the first place, the remaining 6 places are to be occupied by the digits 3, 4, 4, 5, 5, 7 (or 3, 4, 4, 5, 5, 6).

The number of such numbers

$$= \frac{6!}{2!2!} \quad [\because \text{ 4 and 5 each occurs twice}]$$

$$= 180$$

$\therefore$ No. of numbers exceeding 50,00,000 = 360 + 180 + 180 = 720.

■

Example 17

There are 3 *piles of identical red, blue and green balls, where each pile contains at least* 10 *balls. In how many ways can 10 balls be selected:*

(*a*) *if there is no restriction?*

(*b*) *if at least one red ball must be selected?*

(*c*) *if at least one red ball, at least* 2 *blue balls and at least 3 green balls must be selected?*

(*d*) *if exactly one red ball must be selected?*

(*e*) *if exactly one red ball and at least one blue ball must be selected?*

(*f*) *if at most one red ball is selected?*

(*g*) *if twice as many red balls as green balls must be selected?*

Solution

(*a*) There are $n = 3$ kinds of balls and we have to select $r = 10$ balls, when repetitions are allowed.

$\therefore$ No. of ways of selecting = $C(n + r - 1, r) = C(12, 10) = 66$.

(*b*) We take one red ball and keep it aside. Then we have to select 9 balls from the 3 kinds of balls and include the first red ball in the selections.

$\therefore$ No. of ways of selecting = $C(11, 9) = 55$.

(*c*) We take away 1 red, 2 blue and 3 green balls and keep them aside. Then we select 4 balls from the 3 kinds of balls and include the 6 already chosen bolls in each selection.

$\therefore$ No. of ways of selecting = $C(3 + 4 - 1, 4) = 15$.

(*d*) We select 9 balls from the piles containing blue and green balls and include 1 red ball in each selection.

$\therefore$ No. of ways of selecting = $C(2 + 9 - 1, 9) = 10$.

(*e*) We take away one red ball and one blue ball and keep them aside. Then we select 8 balls from the blue and green piles and include the already reserved red and blue balls to each selection.

$\therefore$ No. of ways of selecting = $C(2 + 8 - 1, 8) = 9$.

(*f*) The selections must contain no red ball or 1 red ball.

$\therefore$ No. of ways of selecting = $C(2 + 10 - 1, 10) + C(2 + 9 - 1, 9)$
$= 11 + 10 = 21$.

(*g*) The selections must contain 0 red and 0 green balls or 2 red and 1 green balls or 4 red and 2 green balls or 6 red and 3 green balls.

$\therefore$ No. of ways of selecting = $C(1 + 10 - 1, 10) + C(1 + 7 - 1, 7)$
$+ C(1 + 4 - 1, 4) + C(1 + 1 - 1, 1)$
$= 1 + 1 = 1 + 1 = 4$. ■

Example 18

How many positive integers less than 10,00,000 *have the sum of their digits equal to* 19?

Solution

Any positive integer less than 10,00,000 will have a maximum of 6 digits. If we denote them by x_i, $1 \le i \le 6$, the problem reduces to one of finding the number of solutions of the equation

$$x_1 + x_2 + ... + x_6 = 19, \text{ where } 0 \le x_i \le 9 \qquad ...(1)$$

There are $C(6 + 19 - 1, 19) = C(24, 5)$ solutions if $x_i \ge 0$

We note that one of the six x_i's can be ≥ 10, but not more than one, as the sum of the x_i's = 19.

Let $x_1 \ge 10$ and let $u_1 = x_1 - 10$, $u_i = x_i$, $2 \le i \le 6$

Then the equation becomes

$$u_1 + u_2 + ... + u_6 = 9, \text{ where } u_i \ge 0$$

There are $C(6 + 9 - 1, 9) = C(14, 5)$ solutions for this equations.

The digit which is ≥ 10 can be chosen in 6 ways (*viz.*, it may be x_1, x_2, ..., or x_6).

Hence, the number of solutions of the equation $x_1 + x_2 + ... + x_6 = 19$, where any one $x_i > 10$ is $6 \times C(14, 5)$.

Hence, the required number of solutions of (1)

$$= C(24, 5) - 6 \times C(14, 5)$$
$$= 42{,}504 - 6 \times 2002 = 30{,}492. \quad \blacksquare$$

Example 19

If we select 10 points in the interior of an equilated triangle of side 1. *Show that there must be at least two points whose distance apart is less than* $\frac{1}{3}$.

Solution

Let *ADG* be the given equilateral triangle. The pairs of points *B*, **C**; *E*, *F* and *H*, *I* are the points of trisection of the sides *AD*, *DG* and *GA* respectively. We have divided the triangle *ADG* into 9 equilateral triangles each of side $\frac{1}{3}$.

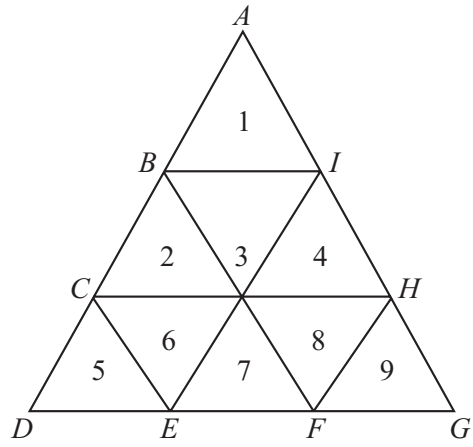


Fig. 2.3

The 9 sub-triangles may be regarded as 9 pigeonholes and 10 interior points may be regarded as 10 pigeons.

Then by the pigeonhole principle, at least one sub triangle must contain 2 interior points.

The distance between any two interior points of any sub triangle cannot exceed the length of the side, namely, $\frac{1}{3}$. $\blacksquare$

Example 20

Find the number of integers between 1 *and* 250 *both inclusive that are not divisible by any of the integers* 2, 3, 5 *and* 7.

Solution

Let *A*, *B*, C, *D* be the sets of integers that lie between 1 and 250 and that are divisible by 2, 3, 5, and 7 respectively.

The elements of A are 2, 4, 6, ..., 250

$\therefore$ $|A| = 125$, which is the same as $\left\lfloor \frac{250}{2} \right\rfloor$

Similarly, $|B| = \left\lfloor \frac{250}{3} \right\rfloor = 83$; $|C| = \left\lfloor \frac{250}{5} \right\rfloor = 50$, $|D| = \left\lfloor \frac{250}{7} \right\rfloor = 35$.

The set of integers between 1 and 250 which are divisible by 2 and 3, *viz.*, $A \cap B$ is the same as that which is divisible by 6, since 2 and 3 are relatively prime numbers.

$\therefore$ $|A \cap B| = \left\lfloor \frac{250}{6} \right\rfloor = 41$

Similarly, $|A \cap C| = \left\lfloor \frac{250}{10} \right\rfloor = 25$; $|A \cap D| = \left\lfloor \frac{250}{14} \right\rfloor = 17$

$|B \cap C| = \left\lfloor \frac{250}{15} \right\rfloor = 16$; $|B \cap D| = \left\lfloor \frac{250}{21} \right\rfloor = 11$

$|C \cap D| = \left\lfloor \frac{250}{35} \right\rfloor = 7$; $|A \cap B \cap C| = \left\lfloor \frac{250}{30} \right\rfloor = 8$

$|A \cap B \cap D| = \left\lfloor \frac{250}{105} \right\rfloor = 5$; $|A \cap C \cap D| = \left\lfloor \frac{250}{210} \right\rfloor = 3$

$|B \cap C \cap D| = \left\lfloor \frac{250}{105} \right\rfloor = 2$; $|A \cap B \cap C \cap D| = \left\lfloor \frac{250}{210} \right\rfloor = 1$

By the Principle of Inclusion–Exclusion, the number of integers between 1 and 250 that are divisible by at least one of 2, 3, 5 and 7 is given by

$$\begin{aligned} |A \cup B \cup C \cup D| &= \{|A| + |B| + |C| + |D|\} - \{|A \cap B| + \ldots \\ &\quad + |C \cap D|\} + \{|A \cap B \cap C| + \ldots \\ &\quad + |B \cap C \cap D|\} - \{|A \cap B \cap C \cap D|\} \\ &= (125 + 83 + 50 + 35) - (41 + 25 + 17 \\ &\quad + 16 + 11 + 7) + (8 + 5 + 3 + 2) - 1 \\ &= 293 - 117 + 18 - 1 = 193 \end{aligned}$$

$\therefore$ Number of integers between 1 and 250 that are not divisible by any of the integers 2, 3, 5 and 7

$$= \text{Total no. of integers} - |A \cup B \cup C \cup D|$$

$$= 250 - 193 = 57.$$ ■

Example 21

A_1, A_2, A_3 and A_4 are subsets of a set U containing 75 elements with the following properties. Each subset contains 28 elements; the intersection of any two of the subsets contains 12 elements; the intersection of any three of the subsets contains 5 elements; the intersection of all four subsets contains 1 element.

(*a*) *How many elements belong to none of the four subsets?*

(*b*) *How many elements belong to exactly one of the four subsets?*

(c) *How many elements belong to exactly two of the four subsets?*

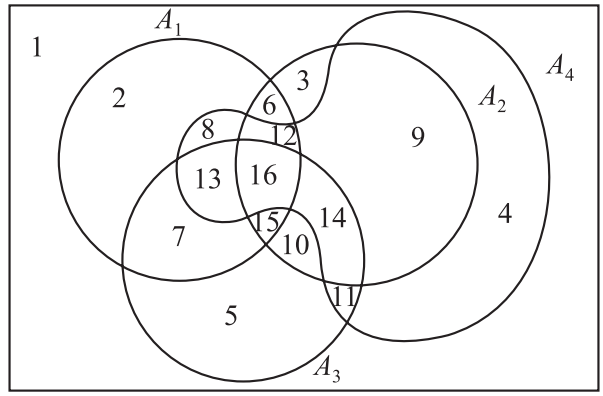


Fig. 2.4

Solution

(*a*) No. of elements that belong to at least one of the four subsets

$$= |A_1 \cup A_2 \cup A_3 \cup A_4|$$
$$= [\{|A_1| + |A_2| + |A_3| + |A_4|\} - \{|A_1 \cap A_2| + |A_1 \cap A_3| + |A_1 \cap A_4| + |A_2 \cap A_3| + |A_2 \cap A_4| + |A_3 \cap A_4|\} + \{|A_1 \cap A_2 \cap A_3| + |A_1 \cap A_2 \cap A_4| + |A_1 \cap A_3 \cap A_4| + |A_2 \cap A_3 \cap A_4|\} - |A_1 \cap A_2 \cap A_3 \cap A_4|]$$
$$= [4 \times 28 - 6 \times 12 + 4 \times 5 - 1] = 59$$

$\therefore$ No. of elements that belong to none of the four subset = 75 – 59 = 16.

(*b*) With reference to the Venn diagram given above Fig. 2.4 $n(A_1$ alone)

$$= n[(2)]$$
$$= n(A_1) - [n(6) + n(7) + n(8) + n(12) + n(13) + n(15) + n(16)]$$

$$= n(A_1) - [\{n(6) + n(12) + n(15) + n(16)\} + \{n(7) + n(13) + n(15) + n(16)\} + \{n(8) + n(12) + n(13) + n(16) - n(12) - n(13) - n(15) - 2n(16)]$$

$$= n(A_1) - [n(A_1 \cap A_2) + n(A_1 \cap A_3) + n(A_1 \cap A_4)] + [n(A_1 \cap A_2 \cap A_4) + n(A_1 \cap A_3 \cap A_4) + n(A_1 \cap A_2 \cap A_3)] - 2n[(A_1 \cap A_3 \cap A_4)]$$

$$= 28 - 3 \times 12 + 3 \times 5 + 2 \times 1$$

$$= 9$$

Similarly $n(A_2$ alone$) = n(A_3$ alone$) = n(A_4$ alone$) = 9$

$\therefore$ No. of elements that belong to exactly one of the subsets = 36.

(*c*) With reference to the Venn diagram of Fig. 2.4 given above, $n(A_1$ and $A_2)$ only $= n(6)$

$$= n(A_1 \cap A_2) - \{n(15) + n(16)\} - \{n(12) + n(16)\} + n(16)$$

$$= n(A_1 \cap A_2) - n(A_1 \cap A_2 \cap A_3) - n(A_1 \cap A_2 \cap A_4) + n(A_1 \cap A_2 \cap A_3 \cap A_4)$$

$$= 12 - 5 - 5 + 1 = 3$$

Similarly $n(A_1$ and A_3 only$) = n(A_1$ and A_4 only$)$

$= n(A_2$ and A_3 only$) = n(A_2$ and A_4 only$)$

$= n(A_3$ and A_4 only$) = 3$

$\therefore$ No. of elements that belong to exactly two of the subsets = 18.

■

Example 22

Show that the number of derangements of a set of n elements is given by

$$D_n = n\,! \left[1 - \frac{1}{1!} + \frac{1}{2!} - \frac{1}{3!} + \ldots + (-)^n \frac{1}{n!}\right]$$

Proof

Let a permutation have the property A_r, if it contains the rth element in the rth position.

Then D_n = the no. of the permutations having none of the properties $A_r\,(r = 1, 2, \ldots, n)$

$$= |\, A_1' \cap A_2' \cap \ldots \cap A_n' \,|$$

$$= N - \sum_i |\, A_i \,| + \sum_{i<j} |\, A_i \cap A_j \,| - \sum_{i<j<k} |\, A_i \cap A_j \cap A_k \,| + \ldots + (-1)^n |\, A_1 \cap A_2 \ldots A_n \,| \qquad \ldots(1)$$

by the principle of inclusion–exclusion, where N is number of permutations of n elements and so equals n !

Now $| A_i | = (n - 1)$!, since $| A_i |$ is the number of permutations in which the ith position is occupied by the ith element, but each of the remaining positions can be filled arbitrarily.

Similarly, $| A_i \cap A_j | = (n - 2)$!, $| A_i \cap A_j \cap A_k | = (n - 3)$! and so on.

Since there are $C(n, 1)$ ways of choosing one element from n, we get

$$\sum_i | A_i | = C(n, 1) \,.\, (n - 1) !$$

Similarly, $\sum_{i<j} | A_i \cap A_j | = C(n, 2) \,.\, (n - 2) !,$

$$\sum_{i<j<k} | A_i \cap A_j \cap A_k | = C(n, 3) \,.\, (n - 3) ! \text{ and so on.}$$

Using these values in (1), we have

$$D_n = n ! - C(n, 1) \,.\, (n - 1) ! + C(n, 2) \,.\, (n - 2) ! - \ldots + (-1)^n \,.\, C(n, n) \,.\, (n - n) ! \qquad \ldots(2)$$

i.e.,
$$D_n = n ! - \frac{n !}{1 ! (n-1) !} (n - 1) ! + \frac{n !}{2 ! (n-2) !} (n - 2) ! - \ldots + (-1)^n \frac{n !}{n ! \, 0 !} 0 !$$

$$= n ! \left\{ 1 - \frac{1}{1!} + \frac{1}{2!} - \frac{1}{3!} + \ldots + (-1)^n \frac{1}{n!} \right\} \quad \blacksquare$$

2.5 The Binomial Theorem

Let x and y be any real numbers, and n any non-negative integer.

Then $(x + y)^n = \sum_{r=0}^{n} C(n, r) \,.\, x^{n-r} \,.\, y^r$

Proof

Since $(x + y)^n = (x + y)\,(x + y)\,(x + y) \ldots (x + y)$ to n factors $(x + y)^n$ is extended by multiplication of x from some of the factors of right hand side and $a\ y$ from the remaining factors.

Every term is obtained by selecting an x from any of the $n - r$ factors and a y from remaining r factors.

Thus, every term of the expansion is of the form $C\, x^{n-r} \,.\, y^r$ where C is the coefficient and $0 \le r \le n$.

We can notice that the coefficient of $x^{n-r} \,.\, y^r$ is the number of ways of selecting an x from any $n - r$ of the n factors. Therefore coefficient of

$$x^{n-r} \,.\, y^r = C(n, n-r) = C(n, r)$$

So, every term in the expansion is of the form

$$C(n, r)\, x^{n-r} \,.\, y^r \quad \text{where} \quad 0 \le r \le n$$

Thus, $$(x + y)^n = \sum_{r=0}^{n} C(n, r)\, x^{n-r}\, y^r$$

An Important Theorem

The set with n elements has 2^n subsets or

$$\sum_{r=0}^{n} C(n, r) = 2^n$$

Proof

From binomial theorem, we have

$$(x + y)^n = \sum_{r=0}^{n} C(n, r)\, x^{n-r}\, y^r$$

Le $$x = y = 1$$

Then $$2^n = (1 + 1)^n = \sum_{r=0}^{n} C(n, r)\, 1^{n-r} \,.\, 1^r$$

$$\Rightarrow \quad \sum_{r=0}^{n} C(n, r) = 2^n. \qquad \blacksquare$$

Example 23

Prove that the sum of 'even' binomial coefficients equals that of the 'odd' binomial coefficients. Where $n > 1$.

Or

$$C(n, 0) + C(n, 2) + C(n, 4) + ... = C(n, 1) + C(n, 3) + C(n, 5) + ...$$

Solution

By binomial theorem,

$$(x+y)^n = \sum_{r=0}^{n} C(n,r)\, x^{n-r}\, y^r$$

Set $x = 1$ and $y = -1$

Then 1 for $x \geq 1$

$$0 = [1 + (-1)]^n = \sum_{r=0}^{n} C(n,r)\,.1^{n-r}\,.(-1)^r$$

$$\Rightarrow \quad 0 = C(n,0) - C(n,1) + C(n,2) - C(n,3) + C(n,4) - C(n,5) + \ldots$$

i.e., $C(n,0) + C(n,2) + C(n,4) + \ldots = C(n,1) + C(n,3) + C(n,5) + \ldots$

■

Example 24

Find the binomial expansion of $(2a - 3b)^4$.

Solution

Here $x = 2a, y = -3b$ and $n = 4$

Using binomial theorem

$$\begin{aligned}(2a-3b)^4 &= C(4,0)\,(2a)^4\,(-3b)^0 + C(4,1)\,(2a)^3\,(-3b)^1 \\ &\quad + C(4,2)\,(2a)^2\,(-3b)^2 + C(4,3)\,(2a)^1\,(-3b)^3 \\ &\quad + C(4,4)\,(2a)^0\,(-3b)^4 \\ &= (2a)^4 + 4(2a)^3\,(-3b) + 6(2a)^2\,(-3b)^2 \\ &\quad + 4(2a)\,(-3b)^3 + (-3b)^4 \\ &= 16a^4 - 96a^3b + 216a^2b^2 - 216ab^3 + 81b^4. \blacksquare\end{aligned}$$

2.6 *n*th Catalan Number

If P_n denotes the number of ways of parenthesizing $(n + 1)$ symbols using n pairs of left and right parentheses.

Then $P_1 = 1$

so, let $n \geq 2$

The first i symbols can be parenthesizing in P_i ways and the next $(n - i)$ symbols in P_{n-i} ways, where $1 \leq i \leq n - 1$.

By using multiplication and addition property, we have

$$P_n = P_1P_{n-1} + P_2P_{n-2} + ... + P_{n-1}\,P_1$$

[from Gegner's formula $P_n = C_{n-1}$]

Considering a generating functions

$$f(x) = P_1x + P_2x^2 + ... + P_nx^n + ...$$

Then
$$|f(x)|^2 = P_1{}^2x^2 + (P_1P_2 + P_2P_1)x^3 + ...$$
$$+ (P_1P_{n-1} + P_2P_{n-2} + ... + P_{n-1}\,P_1)\,x^n + ...$$
$$= x^2 + P_3x^3 + ... + P_nx^n + ...$$
$$= f(x) - x$$

so, $|f(x)|^2 - f(x) + x = 0$

Solving, we get

$$f(x) = \frac{1 \pm \sqrt{1-4x}}{2}$$

Since $P_n > 0$ for every n, we take –ve sign, so

$$f(x) = \frac{1 - \sqrt{1-4x}}{2}$$

Using power series expansion, this gives

$$f(x) = \sum_{n=1}^{\infty} \frac{1}{n} C(2^{n-2}, n-1)x^n$$

Thus,
$$P_n = \frac{1}{n} C(2^{n-2}, n-1) = C_{n-1}$$

so
$$C_n = \frac{1}{n+1} C(2^n, n)\ \forall\ n \geq 0.$$ ■

2.7 Principle of Mathematical Induction (P.M.I)

Mathematical Induction is a technique which is frequent to apply is programming languages and computational methods. The principle of mathematical induction is a consequence of the axioms for Z and N specifically it is a consequence of well ordering principle.

If $P(n)$ be a proposition which is the function of $n \in N$.

To prove $P(n)$ is true for all n, the first step is to show that $P(1)$ is true. We can assume that $P(k)$ (for $k < n$) is true and then we prove that $P(k+1)$ is true. Finally we conclude that $P(n)$, $n \in N$ is true for all n.

Theorem

If $P(n)$ is a proposition where $n \in N$ Satisfying the conditions:

(i) $P(1)$ is true.

(ii) $P(k)$ is true for each $k \in N$, $(1 \le k < n)$ implies $P(k+1)$ is true.

Then it follows that $P(n)$ is true for all $n \in N$.

Example 25

Using mathematical induction prove that for an integer $n \ge 1$.

$$1^3 + 2^3 + 3^3 + \ldots + n^3 = (1 + 2 + \ldots + n)^2$$

Solution

Let $$P_1(n) = 1 + 2 + \ldots n = \frac{n}{2}(n+1)$$

and $$P_2(n) = 1^3 + 2^3 + \ldots + n^3 = \frac{n^2}{4}(n+1)2$$

We shall use mathematical induction to prove that $P_1(n)$ and $P_2(n)$ are true for all $n \in N$. For $n = 1$ we have $P_1(1) = 1$ and R.H.S. = 1 also $P_2(1) = 1$ and R.H.S. = 1. Hence both $P_1(1)$ and $P_2(1)$ are true.

Now we shall assume that $P_1(K)$ and $P_2(K)$ are true, that is

$$1 + 2 + \ldots + K = \frac{K}{2}(K+1)$$

and $$1^3 + 2^3 + \ldots + K^3 = \frac{K^2}{4}(K+1)^2$$

then we shall show that $P_1(K+1)$, $P_2(K+1)$ are true. We have

$$P_1(K+1) = 1 + 2 + \ldots + K + (K+1)$$

$$= \frac{K}{2}(K+1) + (K+1) = \left(\frac{K}{2} + 1\right)(K+1)$$

$$= \left(\frac{K+1}{2}\right)(K+2)$$

Hence $P_1(K+1)$ is true.

Also we have $P_2(K+1) = 1^2 + 2^3 + \ldots + K^3 + (K+1)^3$

$$= \frac{K^2}{4}(K+1)^2 + (K+1)^3$$

$$= \left\{\frac{K^2}{4} + (K+1)\right\}(K+1)^2$$

$$= \frac{1}{4}(K^2 + 4K + 4)\,(K+1)^2$$

$$= \frac{(K+1)^2}{4}(K+2)^2$$

Hence $P_2(K+1)$ is true.

Hence $P_1(n)$ and $P_2(n)$ are true by mathematical induction and from $P_1(n)$ and $P_2(n)$ it is clear that

$$(1 + 2 + ... + n)^2 = 1^3 + 2^3 + ... + n^3.$$ ■

Example 26

Prove that $n^3 + 2^n$ *is divisible by* 3 *for all* $n \geq 1$.

Solution

Let $P(n) = n^3 + 2^n$ is divisible by 3

for $n = 1$

We have $n^3 + 2^n = 1^3 + 2(1) = 3$ is divisible by 3

Hence $P(1)$ is true.

Let $P(k)$ be true *i.e.*, $k^3 + 2k$ is divisible by 3.

We consider,

$$P(k+1) = (k+1)^3 + 2(k+1) = (k+1)\,[(k+1)^2 + 2]$$
$$= k^3 + 3k^2 + 5k + 3$$
$$= (k^3 + 2k) + 3(k^2 + k + 1)$$
$$= 3r + 3(k^2 + k + 1) \text{ where } k^3 + 2k = 3r \text{ (say)}$$
$$= 3(r + k^2 + k + 1)$$

Hence $(k+1)^3 + 2(k+1)$ is divisible by 3 *i.e.*, $P(k+1)$ is true wherever $P(k)$ is true.

Hence by *PMI*, $P(n)$ is true. ■

Example 27

Prove by Principle of mathematical induction the result

$$1 + 3 + 5 + ... + (2n - 1) = n^2.$$

Solution

When $n = 1$, $\Rightarrow$ $1 = 1^2$, the result is true.

Let the result be true for $n = k$ (say)

Then $1 + 3 + 5 + ... + (2k - 1) = k^2$ ...(*i*)

Hence for $n = k + 1$, we get

$$\begin{aligned}\text{L.H.S.} &= 1 + 3 + 5 + ... + (2k - 1) + 2(k + 1) - 1 \\ &= 1 + 3 + 5 + ... + 2k + 1 \\ &= \{1 + 3 + 5 + ... + (2k - 1)\} + (2k + 1) \\ &= k^2 + 2k + 1 \quad \text{(by eqn. (}i\text{))} \\ &= (k + 1)^2 \\ &= \text{R.H.S.}\end{aligned}$$

Therefore the result is true when $n = k + 1$. Since it is true for $n = 1$, it is true for all $n \in N$. ■

2.8 Recurrence Relations

The continuous formation of mathematical relations is a typical task.

Recurrence relations arise in many counting problems and in analysing computer programming problems. When the problem is to find an explicit formula for a recrusively defined sequence, the recursive formula is called a *recurrence relation*. We know that to define a sequence recursively, a recursive formula must be accompanied by information about the beginning of the sequences. This beginning information is called the initial condition or conditions for the sequence.

The general term a_n of a sequence $\{a_n\}$ is often described through a relation connecting a_n with some of its predecessors in the sequence, called a recurrence relation. In some simple cases, an explicit formula for a_n can be obtained by using such a recurrence relation if the first few terms of the sequence $\{a_n\}$ are known before hand. For example, if a_1 is known and a_n is expressed in terms of a_{n-1} through a recurrence relation, then an explicit formula for a_n can be obtained by using what is called the *backtracking technique*. This technique, consists in substituting for an in terms of a_{n-1}, a_{n-1} in terms of a_{n-2}, a_{n-2} in terms of a_{n-3}, and so on, and finally obtaining a relation that gives a_n in terms of a_1 which is known.

The backtracting technique is very tedious if the recurrence relation expresses a_n in terms of two or more of its predecessors. In such situations, we employ what is known as the different equation technique.

Suppose that the recurrence relation is of the form

$$a_n = k_1 a_{n-1} + k_2 a_{n-2},\ n \geq 3 \qquad ...(1)$$

and that a_1 and a_2 are specified as

$$a_1 = a,\ a_2 = b \qquad ...(2)$$

where k_1, k_2, a, b are known constants.

A relation of the form (1) is called a *linear homogeneous difference equation of degree two*. The relations of the form (2) which specify a_1 and a_2 are called the *initial conditions*.

Given the relation (1), we first write down the following quadratic equation, known as the *characteristic equation* (or *auxiliary equation*) associated with the relation (1):

$$x^2 = k_1 x + k_2$$

or
$$x^2 - k_1 x - k_2 = 0 \qquad ...(3)$$

Observe that this equation has been written down by replacing a_n by x^2, a_{n-1} by x and and a_{n-2} by 1, in the relation (1).

Solving equation (3), we obtain two roots α_1 and α_2 (say) which may or may not be distinct. If α_1 and α_2 are *distinct*, then we infer that a_n is of the form

$$a_n = A\alpha_1^n + B\alpha_2^n \qquad ...(4)$$

and if $\alpha_1 = \alpha_2 = a$ (say), we infer that a_n is of the form

$$a_n = (A + nB)a_n \qquad ...(5)$$

Here, A and B are constants to be determined. Since a_1 and a_2 are known before hand through the initial conditions (2), the constants A and B can be evaluated by using the specified values of a_1 and a_2 and the expression (4) or (5), as the case may be.

An explicit formula for a_n may be arrived at.

This method of obtaining an explicit formula for a_n is analogous to the method of solving an ordinary second-order, linear homogeneous differential equation with constant coefficients.

An Important Theorem

If the characteristic equation $x^2 - r_1 x - k_2 = 0$ of the recurrence relation $a_n = r_1 a_{n-1} + r_2 a_{n-2}$ has two distinct roots s_1 and s_2 then $a_n = Us_1^n + Vs_2^n$ where U and V depend on the initial conditions, is the explicit formula for the sequence.

Proof

Let $s_1^2 - r_1 s_1 - r_2 = 0$ ...(*i*)

$s_2^2 - r_1 s_2 - r_2 = 0$ ...(*ii*)

and $a_n = Us_1 n + Vs_2 n$ for $n > 1$ ...(*iii*)

Now, we need to show that this definition of a_n defines the same sequence as $a_n = r_1 a_{n-1} + r_2 a_{n-2}$. First let us assume that U and V are so choosen that $a_1 = Us_1 + Vs_2$ and $a_2 = Us_1^2 + Vs_2^2$ and the initial conditions are satisfied. Then

$$\begin{aligned} a_n &= Us_1^n + Vs_2^n \\ &= Us_1^{n-2} \cdot s_1^2 + Vs_2^{n-2} \cdot s_2^2 \\ &= Us_1^{n-2}(r_1 s_1 + r_2) + Vs_2^{n-2}(r_1 s_2 + r_2) \text{ using } (i) \text{ and } (ii) \\ &= r_i Us_1^{n-1} + r_2 Us_1^{n-2} + r_1 Vs_2^{n-1} + r_2 Vs_2^{n-2} \\ &= r_1(Us_1^{n-1} + Vs_1^{n-1}) + r_2(Us_1^{n-2} + Vs_2^{n-2}) \end{aligned}$$

i.e., $a_n = r_i\, a_{n-1} + r_2\, a_{n-2}$ using (*iii*) in definitions of a_{n-1} and a_{n-2}.

■

Example 28

Akshita deposits ₹ 1000 *in a local saving bank at annual interest rate of* 8% *compound annually. Define recursively the compound amount* $A(n)$ *she will have in her account at the end of n years.*

Solution

It is clear that, $A(0)$ = initial deposit = ₹ 1000.

Let $n \geq 1$. Then

$$A(n) = \begin{pmatrix} \text{Compounded amount} \\ \text{at the end of the} \\ (n-1) \text{ years} \end{pmatrix} + \begin{pmatrix} \text{Interest earned} \\ \text{during the} \\ n\text{th year} \end{pmatrix}$$

$$= A(n-1) + (0.08)\, A(n-1)$$

$$= 1.08\, A(n-1)$$

Thus $A(n)$ can be defined recursively as follows:

$A(0) = 1000$ ← initial condition

$A(n) = 1.08\, A(n-1), \forall\ n \geq 1$ ← recurrence relation

Illustration

For instance, the compound amount will be at the end of three years is

$$\begin{aligned} A(3) &= 1.08\, A(2) \\ &= 1.08[1.08\, A(1)] \\ &= (1.08)^2\, A(1) \\ &= (1.08)^2\, [1.08\, A(0)] = (1.08)^3\, (1000) \\ &\approx ₹\ 1259.71. \end{aligned}$$

■

Example 29

*(**The handshake problem**) There are n guests in a function hall. Each person shakes hands with everybody else exactly once. Define recursively the number of handshakers h(n) that occur.*

Solution

It is clear that, $h(1) = 0$, so let $n \geq 2$.

Let x be one of the guests.

By definition, the number of handshakes made by the remaining $(n-1)$ guests among themselves is $h(n-1)$.

Now person x shakes hands with each of these $(n-1)$ guests, yielding $(n-1)$ additional handshakes.

The total number of handshakes made equals $h(n-1) + (n-1)$, where $n \geq 2$.

Thus $h(n)$ can be defined recursively as following:

$h(1) = 0$ ← initial condition

$h(n) = h(n-1) + (n-1)$ ← recurrence relation

To solve this recurrence relation

Step 1: To predict a formula $h(n)$:

Using iteration,

$$\begin{aligned} h(n) &= h(n-1) + (n-1) \\ &= h(n-2) + (n-2) + (n-1) \\ &= h(n-3) + (n-3) + (n-2) + (n-1) \\ &\quad \vdots \\ &= h(1) + 1 + 2 + 3 + \ldots + (n-2) + (n-1) \\ &= 0 + 1 + 2 + 3 + \ldots + (n-1) \\ &= \frac{n(n-1)}{2}. \end{aligned}$$

Step 2: To prove, by induction, that $h(n) = \frac{n(n-1)}{2}$, $n \geq 1$

Basic Step: When $n = 1$, $h(1) = \frac{1.0}{2} = 0$ this agrees with the initial condition so formula holds when $n = 1$.

Induction Step: Assume $h(k) = \frac{h(k-1)}{2}$ for $k \geq 1$

Then $h(k + 1) = h(k) + k$ (by recurrence relation)

$$= \frac{h(k-1)}{2} + k \quad \text{(induction bypothesis)}$$

$$= \frac{h(k+1)}{2}$$

Therefore, the formula holds for $n = k$, it also holds for $n = k + 1$.

Thus, by *PMI*, the result holds for $n \geq 1$. ■

Example 30

*(**Tower of Brahma or Tower of Hanoi Problem**) According to legend of India, at the beginning of creation, God stacked* 64 *golden disks on of the three diamond pegs on a brass platform is the temple of Brahma at Banaras.*

The priests on duty were asked to move the disks from peg X to peg Z using Y as an auxilary peg under the following conditions:

- *Only one disk can be moved at a time.*
- *No disk can be placed on the top of a smaller disk.*

The priests were told that the world would end when the job was completed.

Solution

Suppose there are n disks on peg X.

Let b_n denotes the number of moves needed to move them from peg X to peg Z, using peg Y as an intermediately and define b_n recursively.

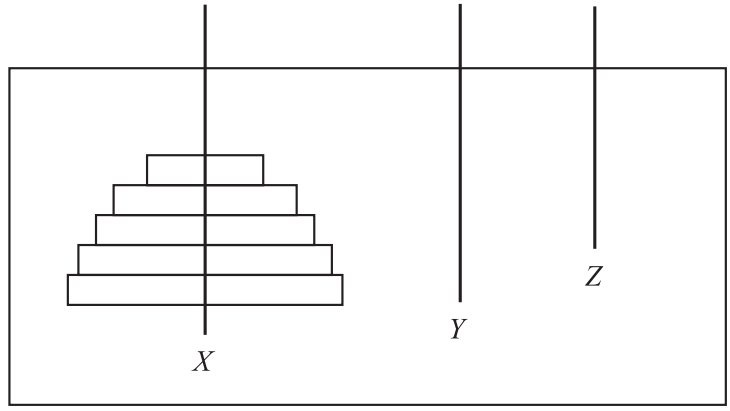


Fig. 2.5: Tower of Brahma problem

It is clear that $b_1 = 1$

Assume $n \geq 2$

We consider the top $(n - 1)$ disks on peg X.

By definition, it takes b_{n-1} moves to transfer them from X to Y using Z as an auxiliary. That leaves the largest disk, at peg X, it takes one move to transfer it from X to Z.

Now $(n - 1)$ disks at Y can be moved from Y to Z using X as an intermediary in b_{n-1} moves, so the total number of moves needed is $b_{n-1} + 1 + b_{n-1} = 2\, b_{n-1} + 1$.

Thus, b_n can be refined recursively as follows:

$$b_n = \begin{cases} 1 \text{ if } n = 1 \longleftarrow \text{initial condition} \\ 2b_{n-1} + 1 \text{ otherwise} \longleftarrow \text{recurrence relation} \end{cases}$$

■

Illustration

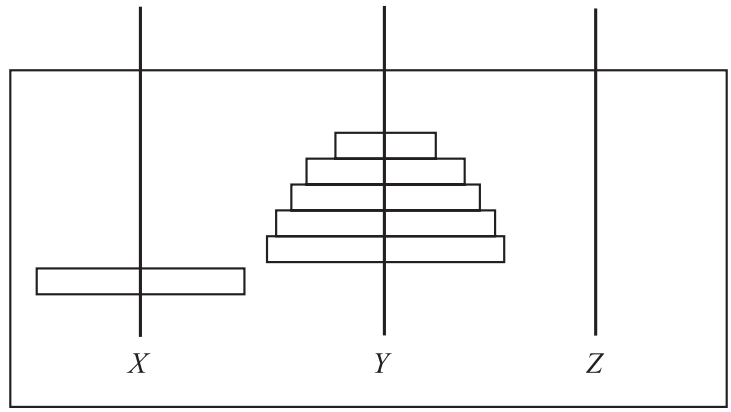


Fig. 2.6

For example,

$$\begin{aligned} b_4 &= 2b_3 + 1 = 2[2b_2 + 1] + 1 \\ &= 4b_2 + 2 + 1 = 4[2b_1 + 1] + 2 + 1 \\ &= 8b_1 + 4 + 2 + 1 = 8(1) + 4 + 2 + 1 \\ &= 15 \end{aligned}$$

so it takes 15 moves to transfer 4 disks from X to Z by using this strategy.

■

Example 31

*(**The Circle Problem**) Let g_n denotes the maximum number of non-overlapping regions formed inside a circle by joining n distinct points on it. Derive a formula for g_n.*

Solution

We derive an explicit formula for g_n, using recursion. Let $P_1, P_2, ..., P_n$ be n points on a circle such that no three chords $\overline{P_rP_s}$ are concurrent, where $1 \le r, s \le n$. Choose a new point Q on arc $\overset{\frown}{P_{n-1}P_n}$. Join Q to each of the points P_1 through P_n. The chord $\overline{QP_n}$ introduces an extra region.

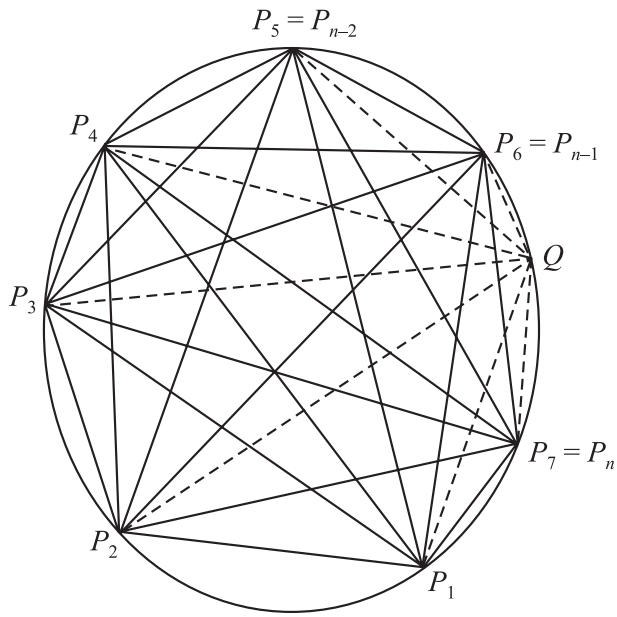


Fig. 2.7

Now consider the chord $\overline{QP_i}$, $1 \le i \le n-1$. It intersects each of the chords obtained by joining one of the i points $P_n, P_1, ..., P_{i-1}$ to one of the $n-1-i$ points $P_{i+1}, P_{i+2}, ..., P_{n-1}$. Each intersection corresponds to a new region.

Consequently, the number of regions formed by the chord $\overline{QP_i}$ is one more than the total number of points of intersection, namely, $1 + i(n-1-i)$, $1 \le i \le n-1$.

Thus, the total number of regions formed by the introduction of the $(n+1)$st point Q is point Q is

$$1 + \sum_{i=1}^{n-1} [1 + i(n-1-i)] = \left(\sum_{i=1}^{n} 1 \right) + \sum_{i=1}^{n-1} [i(n-1-i)]$$

This yields the recurrence relation

$$g_{n+1} - g_n = \left(\sum_{i=1}^{n} 1 \right) + \sum_{i=1}^{n-1} [i(n-1-i)]$$

The R.H.S. of this equation can be simplified as follows:

$$= n + \sum_{i=1}^{n-1} i(n-1) - \sum_{i=1}^{n-1} i^2$$

$$= n + (n-1) \sum_{i=1}^{n-1} i - \sum_{i=1}^{n-1} i^2$$

$$= n + (n-1).\frac{(n-1)n}{2} - \frac{(n-1)n[2(n-1)+1]}{6}$$

$$= n + \frac{n(n-1)^2}{2} - \frac{n(n-1)(2n-1)}{6}$$

$$= n + \frac{n(n-1)(n-2)}{6}$$

$$= C(n, 1) + C(n, 3) \quad \text{(we can verify this)}$$

Solving this recurrence relation by iteration yields

$$g_n = C(n, 0) + C(n, 2) + C(n, 4), \quad n \ge 0.$$

In particular, $g_5 = C(5, 0) + C(5, 2) + C(5, 4) = 1 + 10 + 5 = 16$

and $g_6 = C(6, 0) + C(6, 2) + C(6, 4) = 1 + 15 + 15 = 31$,

as expected. ■

Example 32

Consider the recurrence relation $a_n = a_{n-1} + 4$ *with* $a_1 = 2$ *which defines the sequence* 2, 6, 10, 14 ...

Solution

Now we backtrack the value of a_n by using the definition of a_{n-1}, a_{n-2} and so on until a pattern is clear

$$a_n = a_{n-1} + 4 \quad \text{or} \quad a_n = a_{n-1} + 4$$
$$= (a_{n-2} + 4) + 4 = a_{n-2} + 2.4$$
$$= [(a_{n-3} + 4) + 4] + 4 = a_{n-3} + 3.4$$

Continuing this process, will produce

$$a_n = a_{n-(n-1)} + (n-1) \,.\, 4$$
$$= a_1 + (n-1)\, 4$$

i.e.,
$$a_n = 2 + (n-1)\, 4$$

Thus an explicit formula for the given sequence is $a_n = 2 + (n-1)\, 4$.

■

Example 33

Solve the recurrence relation

$$a_n = 2a_{n/2} + (n-1),\ n \geq 2 \text{ with } a_1 = 0$$

for $n = 2k$ *where* $k \geq 1$.

Solution

When $n \geq 2$, we have (by what is given)

$$a_n = 2a_{n/2} + (n-1)$$

From this, we find that

$$2a_{n/2} = 2\left\{2a_{n/4} + \frac{n}{2} - 1\right\}$$
$$= 2^2 a_{n/4} + (n-2)$$
$$2^2 a_{n/4} = 2^2\left\{2a_{n/8} + \frac{n}{4} - 1\right\}$$
$$= 2^3 a_{n/8} + (n - 2^2)$$

and so on.

Thus, we have the following expressions:

$$a_n - 2a_{n/2} = (n-1)$$
$$2a_{n/2} - 2^2 a_{n/4} = (n-2)$$
$$2^2 a_{n/4} - 2^3 a_{n/8} = (n-2^3)$$
.....
.....
$$2^{k-1} a_{n/2^{k-1}} - 2k_{an/2^k} = (n-2^{k-1})$$

Adding all these expressions, we get

$$a_n - 2^k a_{n/2^k} = (n-1) + (n-2) + (n-2^2) + \ldots + (n-2^{k-1})$$

For $n = 2^k$, we have $a_{n/2^k} = a_1 = 0$ (by what is given. Therefore (for $n = 2^k$),

$$\begin{aligned} a_n &= (n + n + \ldots + n) - (1 + 2 + 2^2 + \ldots + 2^{k-1}) \\ &\qquad (k \text{ terms}) \\ &= kn - (2^k - 1) \\ &= (\log_2 n)\, n - (n-1), \text{ because } n = 2^k \\ &= 1 + n(\log_2 n - 1) \end{aligned}$$

Thus the given recurrence relation is solved completely (for $n = 2k, k \geq 1$).

■

Example 34

Use the technique of backtracking to find an explicit formula for the sequence defined by

$$a_n = a_{n-1} + n, \quad a_1 = 4$$

Solution

By backtracking, we find that

$$\begin{aligned} a_n &= a_{n-1} + n \\ &= \{a_{n-2} + (n-1)\} + n \\ &= \{[a_{n-3} + (n-2)] + (n-1)\} + n \\ &= a_{n-3} + (n-2) + (n-1) + n \\ &= a_{n-4} + (n-3) + (n-2) + (n-1) + n \end{aligned}$$
.....
.....
$$= a_1 + 2 + 3 + 4 + \ldots + (n-1) + n$$

$$= (a_1 - 1) + [1 + 2 + 3 + 4 + ... + (n-1) + n]$$

$$= 3 + \frac{n(n+1)}{2}$$

on using the given value of a_1 and the standard result

$$1 + 2 + 3 + ... \; n = \frac{1}{2}n(n+1)$$

Thus, we have $a_n = \frac{1}{2}n(n+1) + 3$

This is the explicit formula for the given sequence. ■

Example 35

Find a formula for the general term F_n of the Fibonacci sequence 0, 1, 1, 2, 3, 5, 8, 13, ...

Solution

The recurrence relation corresponding to the Fibonacci sequence $\{F_n\}$; $n \geq 0$ is $F_{n+2} = F_{n+1} + F_n$; $n \geq 0$ with the initial conditions $F_0 = 0$, $F_1 = 1$.

The characteristic equation of the R.R. is

$$r^2 - r - 1 = 0$$

Solving it, we have $r = \frac{1 \pm \sqrt{5}}{2}$

Since the R.S. of $F_{n+2} - F_{n+1} - F_n = 0$ is zero, the solution of the R.R. is

$$F_n = C_1\left(\frac{1+\sqrt{5}}{2}\right)^n + c_2\left(\frac{1-\sqrt{5}}{2}\right)^n$$

$F_0 = 0$ gives $c_1 + c_2 = 0$ ...(1)

$$F_1 = 1 \text{ gives } c_1\left(\frac{1+\sqrt{5}}{2}\right) + c_2\left(\frac{1-\sqrt{5}}{2}\right) = 1 \qquad ...(2)$$

Using (1) in (2), we get $c_1 - c_2 = \frac{2}{\sqrt{5}}$ ...(3)

Using (1) in (3), we have $c_1 = \frac{1}{\sqrt{5}}$ and $c_2 = \frac{1}{\sqrt{5}}$

$\therefore$ The general term F_n of the Fibonacci sequence is given by

$$F_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{n}\right)^n; \; n > 0.$$ ■

SUMMARY

1. The Fundamental Counting Principles

- **Addition Principle:** If A and B are two mutually exclusive tasks and can be done in m and n ways, respectively, task A or B can be done in $m + n$ ways.
- **Inclusion–Exclusion Principle:** Suppose task A can be done in m ways and task B in n ways. If both can be done in k ways, task A or B can be done in $m + n - k$ ways.
- **Multiplication Principle:** If task T_1 can be done in m_1 ways and task T_2 in m_2 ways corresponding to each way T_1 can occur, these two tasks can be done in that order in $m_1 m_2$ ways.

2. Permutations

- An r-permutation of a set of n distinct elements is an ordered arrangement of r elements of the set. The number of r-permutations of a set of size n is denoted by $P(n, r)$.
- $P(n, r) = \dfrac{n!}{(n-r)!}$, $P(n, n) = n!$
- A cyclic permutation is a circular arrangement. The number of cyclic permutations of n distinct items is $(n - 1)!$
- $P(n, r) = P(n - 1, r) + rP(n - 1, r - 1)$.

3. Derangements

- A **derangement** is a permutation of n distinct items $a_1, a_2, ..., a_n$ such that no item a_i occupies position i, where $1 \le i \le n$.
- The number of derangements D_n of n items satisfies two recurrence relations:

  $D_n = (n - 1)(D_{n-1} + D_{n-2})$, $n > 2$ where $D_0 = 1$ and $D_1 = 0$

  $D_n = nD_{n-1} + (-1)^n$, $n \ge 1$ where $D_0 = 1$.

- $D_n = n!\left[1 - \dfrac{1}{1!} + \dfrac{1}{2!} ... + \dfrac{(-1)^n}{n!}\right]$, $n \ge 0$.

4. Combinations

- An r-combination of a set of n elements is a subset with size r, where $0 \le r \le n$. The number of r-combinations is denoted by $C(n, r)$ or $\binom{n}{r}$.

- $C(n, r) = \dfrac{n!}{r!(n-r!)}, \quad C(n, r) = C(n, n-r).$

- **Pascal's identity** $C(n, r) = C(n-1, r-1) + C(n-1, r)$.

- **Permutations with Repetitions:** The number of permutations of n items where n_1 are alike, n_2 are alike, ..., and n_k are alike, is given by $\dfrac{n!}{n_1!\, n_2! \ldots n_k!}$.

- **Combinations with Repetitions:** The number of r-combinations with repetitions from a set with size n is $C(n + r - 1, r)$.

- Let $x_1, x_2, \ldots, x_n$ be n non-negative integer variables. The equation $x_1 + x_2 + \ldots + x_n = r$ has $C(n + r - 1, r)$ integer solutions.

- **The binomial theorem:** Let x and y be real variables and n any non-negative integer. Then

$$(x + y)n = \sum_{r=0}^{n} \binom{n}{r} x^{n-r}\, y^r.$$

- $\displaystyle\sum_{r=0}^{n} C(n, r) = 2^n.$

- The number of surjections N from a finite set A to a finite set B is given by

$$N = \sum_{r=0}^{n} (-1)^r \binom{n}{r} (n-r)^m \quad \text{where } |A| = m, |B| = n, \text{ and } m \ge n.$$

- The number of derangements N of n items is given by

$$N = \sum_{r=0}^{n} (-1)^r \binom{n}{r} (n-r)!.$$

EXERCISE

1. How many bit strings of length 10 contain

(*a*) exactly four 1's

(*b*) almost four 1's

(*c*) at least four 1's

(*d*) an equal number of 0's and 1's?

[**Ans.** (*a*) 210, (*b*) 386, (*c*) 848, (*d*) 252]

2. If 6 people *A*, *B*, *C*, *D*, *E*, *F* are seated about a round table, how many different circular arrangements are possible, if arrangements are considered the same when one can obtained from the other by rotation? [**Ans.** 120, 6, 2, 12]

3. From a club consisting of 6 men and 7 women, in how many ways can we select a committee of

(*a*) 3 men and 4 women?

(*b*) 4 persons which has at least one women?

(*c*) 4 persons that has at most one man?

(*d*) 4 persons of both sexes?

(*e*) 4 persons so that two specific members are not included?

[**Ans.** (*a*) 700, (*b*) 700, (*c*) 245, (*d*) 665, (*e*) 660]

4. Find the number of non-negative integer solutions of the iniquality $x_1 + x_2 + x_3 + x_4 + x_5 + x_6 < 10$? [**Ans.** 5005]

5. Prove, by mathematical Induction, that

$$1^1 + 3^2 + 5^2 + \ldots + (2n-1)^2 = \frac{1}{3}n(2n-1)(2n+1).$$

6. Prove, by mathematical Induction, that

$$1 + \frac{1}{1.2} + \frac{1}{2.3} + \frac{1}{3.4} + \ldots + \frac{1}{n(n+1)} = \frac{n}{n+1}.$$

7. Use mathematical induction to show that

$n\,! > 2^{n-1}$ for $n = 1, 2, 3, \ldots$

8. Use mathematical induction to show that

$$\frac{1}{\sqrt{1}} + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{3}} + \ldots + \frac{1}{\sqrt{n}} > \sqrt{n}, \text{ for } n \geq 2.$$

9. Use mathematical induction to show that

$$\frac{1 \cdot 3 \cdot 5 \ldots (2n-1)}{2 \cdot 4 \cdot 6 \ldots (2n)} \leq \frac{1}{\sqrt{n+1}} \quad \text{for } n = 1, 2, 3, \ldots$$

10. Using mathematical induction to prove that
$n^3 + (n+1)^3 + (n+2)^3$ is divisible by 9, for $n \geq 1$.

11. Solve the recurrence relation
$a_{n+1} - a_n = 3n^2 - n;\ n \geq 0, a_0 = 3.$ [**Ans.** $a_n = 3 + n(n-1)^2$]

12. Solve the recurrence relation
$a_{n+2} - 6a_{n+1} + 9a_n = 3(2^n) + 7(3^n), n \geq 0$
Given that $a_0 = 1$ and $a_1 = 4$

[**Ans.** $a_n = \frac{5}{8} n \cdot 3^n + 2^n + \frac{7}{18} n^2 \cdot 3^n$]

13. Use the method of generating function to solve the recurrence relation
$a_n = 3a_{n-1} + 1$; $n \geq 1$, given that $a_0 = 1$

[**Ans.** $a_n = \frac{1}{2}(3^{n+1} - 1)$]

14. Prove that
$C(n, r) = C(n-1, r-1) + C(n-1, r)$, where $0 < r < n$.

15. Find the explicit formula for the sequence defined by
$a_n = 4a_{n-1} + 5a_{n-2}$; $a_1 = 2, a_2 = 6.$ [**Ans.** $a_n = \frac{4}{15} \cdot 5^n - \frac{2}{3}(-1)^n$]

16. Solve the recurrence relation
$a_n = -6a_{n-1} - 9a_{n-2}$, given that $a_1 = 2.5, a_2 = 4.7$.

[**Ans.** $a_n = \left(\frac{19.7}{9} + \frac{12.2}{9} n\right)(-3)^n$]

Suggested Readings

1. G. Berman and K.D. Fryer, *Introduction to combinatorics*. Academic press, New York, 1972, pp. 35–125.
2. K.P. Bogart, Introductory Combinatorics, Pitman, Boston, MA, 1983.
3. A. Iucker, Applied Combinatorics, Wiley, 2002.

❑❑❑

3

Mathematical Logic

Baron Gottfried Wilhelm Leibnitz (1646–1716)

Baron Gottfried Wilhelm Leibnitz (1646–1716) was an outstanding German mathematician, philosopher, physieist, diplomat and linguist. He was born in a Lutheran family. As the son of a professor of philosophy, he grew up to be a genius with encyclopedic knowledge. He learnt himself Latin, Greek and philosophy before entering the university of Leipzig at 18 as a law student. He read about the works of great scientists and philosophers such as Galileo, Francis Bacon and others. Because he was so young that he was not awarded by Doctor of law degree, so he left his native place forever.

During 1663–1666 he rushed to the universities of Jena and Altdorf, and received his doctorate in law and started legal practice for the Elector of Mainz. After he death of Elector, he pursued scientific studies. In 1672 he built a calculation that could multiply and divide and presented it to the Royal Society in London.

In 1675, Leibnitz laid the foundations of calculus, an honour he shares with Sir Isaac Newton. He discovered the fundamental theorem of calculus, and invented the popular notations d/dx for differentiation and $\int$ for integeration. He also introduced such modern notations as dot for multiplication and decimal point, the equal sign and colon for ratio.

An unmarried fellow, Leibnitz also worked for theology, philosophy, mathematics, natural Science and Technology.

3.1 Introduction

The study of the principles and techniques of reasoning is called logic. It was originated by ancient Greeks, led by the philosopher Aristotle,

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_3

who is called the father of logic. It was not until the 17th century that symbols were used in development of logic. German philosopher and mathematician Gottfried Leibnitz introduced symbolism into logic. No significant contributions in symbolic logic were made until those of George Book, an English mathematician. At the age of 39, Book published his outstanding work in symbolic logic. An investigation of the laws of thought.

It is the logic which plays a central role in the every area of learning basically in mathematics and computer science. computer scientists employ logic to develop programming languages and to establish the correctners of the programs. Electronic Engineers apply logic in the design of computer chips.

Logic is the discipline which deals with methods of reasoning. On an elementary level, logic provides rules and techniques for determining whether a given argument is valid. Logical reasoning is used in mathematical sciences to provide the theorems, in computer science to verify the correctness of programs and to prove theorems, in the natural and physical sciences to draw conclusions from experiment, and in the social sciences and in our everyday life to solve a multitude of problems. Indeed, we are constantly using logical reasoning. Logic is concerned with all kinds of reasonings, whether they be legal arguments or mathematical proofs or conclusions in a scientific theory based on a set of type thesises. Because of the diversity of their application, these rules, called inference, must be stated in general terms and must be independent of any particular argument of any discipline involved.

A collection of rules or any theory needs a language in which these rules or theory can be stated. natural language is not always preucise enough. They are also ambiguous and as such, are not suitable for this purpose. It is therefore necessary first to develop a formal language called the *object language*. A formal language is one in which the syntax is well defined. In fact, every scientific discipline develops its own object language which consists of certain well defined terms and well specified uses of these terms. In order to avoid ambiguity, we use symbols which have been clearly defined in the object languages. An additional reason to use symbols is that they are easy to write and manipulate. Because of this use of symbols, the logic that we shall study is also called *symbolic logic*. Our study of object language requires the use of another language. For this purpose we choose any of the natural languages. In this case our choice is English, and so the statements about the object language will be made

in English. This language will then be called our metalanguage. Certain inherent difficulties in this procedure could be anticipated, because we wish to study a precise language while using another language which is not so precise.

3.2 Propositions (Statements)

A declarative sentence is called statement or proposition which has one and only one of two possible values, called **truth values**. These two values are called **true** and **false** denoted by symbols T and F respectively sometimes these are also denoted by the symbol 1 and 0. We assume the object language contains such declarative sentences. We allow only two possible truth values is our studies called **two valued logic**. Declarative sentences in the object language are of two types. The first type include those sentences which are considered to be prime time in the object language. These are to be denoted by distinct symbols selected from the capital letters A, B, C, ..., P, Q, R ... X, Y, Z, *etc.*, while declarative sentences of the second type are obtained from the primitive once by using certain symbols, called connectives and certain punctuation marks such as parentheses sentences.

Some examples of statements are like.

(i) Pakistan is a country.

(ii) New Delhi is the capital of Nepal.

(iii) This is a true statement.

(iv) 10 + 101 = 110.

(v) Shut the door.

(vi) Pataliputra is an old city.

(vii) Man will reach on marsh by 2020.

It is obvious that the statements (i) and (ii) have truth value **true** and **false** respectively, (iii) is not a statement. In (iv) we have a statement whose truth value depends on the context. We will be interested only is the fact that it has a truth value. In this sense (v), (vi), (vii) are all statements.

3.3 Connectives

When P and Q are any two propositions, the proposition P and Q denoted by $P \wedge Q$ and called the conjunction of P and Q is defined as the compound proposition that is true for both P and Q are true and is false otherwise.

In the case of simple statements, their truth values are fairly obvious. It is possible to construct rather complicated statements from simple statements by using certain connecting words or expressions known as "**sentential connectives**".

The statements which are to be considered initially are simple statements are called atomic or primary statements. New statements can be formed from atomic statements through the use of sentential connectives. The resulting statement are called molecular or compound statements. This shows that atomic statements are those which do not have any connectives.

The main connectives are **negation**, **conjuction** and **disjuction**. Which are related to English words **not**, **and**, **or** respectively.

These are shown as under:

Connective English Word	Name of Connective	Symbol
Not	Negation	$\neg$
And	Conjuction	$\wedge$
Or	Disjunction	$\vee$

3.3.1 Negation

This statement is generally formed by introducing the word "**Not**" at a proper place is the statement or prefixing the statement by phrase "**It is not the case that**". If P denotes a statement, then the negation of 'P' is writters as '$\neg P$' and read as 'not P'. If the truth value of 'P' is T, then the truth value of '$\neg P$' is T.

We consider the statement:

P: Agra is a city.

Than $\neg P$ is the statement.

Than $\neg P$ is the statement that

$\neg P$: It is not case that Agra is a city.

Generally, $\neg P$: Agra is not a city.

The above two statements "It is not the case that Agra is a city" and "Agra is not a city" are not identical. We can translate both of them by $\neg P$.

The reason is that both of these statements are of the same meaning in English.

Truth table for negation

P	¬P
T	F
F	T

Note: The symbol '¬' has to be used here to denote negation.

Alternatively used symbols are "~", a bar or "NOT" so that ¬*P* can be written as ~*P*, $\overline{P}$ or NOT *P*.

3.3.2 Conjunction

The conjunction is the joining of two statements *P* and *Q* and denoted by $P \wedge Q$ which has truth value *T* wherever both *P* and *Q* have the truth value *T* otherwise the truth value *F*.

Truth Table for Conjuction

P	Q	P ∧ Q
T	T	T
T	F	F
F	T	F
F	F	F

For example, if

P: It is very cold.

Q: There are 50 students in my class.

Then $P \wedge Q$: It is very cold and there are 50 students is my class.

3.3.3 Disjunction

The disjunction of two statements *P* and *Q* is to operate the sentences as "*P* or *Q*" which is written as "$P \vee Q$" has the truth value *F* only when both *P* and *Q* have the truth value *F*; otherwise it is true.

Truth table for disjunction

P	Q	P ∨ Q
T	T	T
T	F	T
F	T	T
F	F	F

Combined truth table for negation, conjunction and disjunction

P	Q	⌝P	P ∧ Q	P ∨ Q
T	T	F	T	T
F	T	T	F	T
T	F	F	F	T
F	F	T	F	F

3.3.4 Conditional

If P and Q be any two statements. Then the statement "$P \Rightarrow Q$" read as "If P then Q" or "P implies Q" is termed as conditional statement. The truth value of "$P \Rightarrow Q$" is F when Q has truth value F and P has truth value T.

In all other cases "$P \Rightarrow Q$" has truth value T. P is called the antecedent and Q the consequent in "$P \Rightarrow Q$". The truth table for "$P \Rightarrow Q$" is as under.

Truth table for conditional (P ⇒ Q)

P	Q	P ⇒ Q
T	T	T
T	F	F
F	T	T
F	F	T

3.3.5 Biconditional

If we have two statements P and Q, then the statements $P \Leftrightarrow Q$ is called a biconditional. This we read as "P if and only if Q" and abbreviated as "P iff Q". In general mathematical language we address it as "P is necessary and sufficient of Q" and also "Q is necessary and sufficient of P". $P \Leftrightarrow Q$ has the truth value T whenever both P and Q have identical truth values.

Truth table for biconditional (P ⇔ Q)

P	Q	P ⇔ Q
T	T	T
T	F	F
F	T	F
F	F	T

Note: (*i*) For P and Q, the statement '$Q \Rightarrow P$' is called the converse of $P \Rightarrow Q$.

(*ii*) $\neg P \Rightarrow \neg Q$ is called the inverse of $P \Rightarrow Q$.

(*iii*) $\neg Q \Rightarrow \neg P$ is called the contrapositive of "$P \Rightarrow Q$".

Combined Truth Tables

Table 1

P	Q	P ⇒ Q	Q ⇒ P	⌝P	⌝Q	⌝P ⇒ ⌝Q	⌝Q ⇒ ⌝P
T	T	T	T	F	F	T	T
T	F	F	T	F	T	T	F
F	T	T	F	T	F	F	T
F	F	T	T	T	T	T	T

Table 2

P	Q	P ⇒ Q	Q ⇒ P	P ⇔ Q
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

3.4 Equivalence of Formulae

The two statement formulae A and B in the variables $P_1, P_2, ..., P_n \forall n \geq 1$ are called equivalent if and only if they acquire the same truth values for all interpretation or the possess identical truth values. Two statement formulae which are equvalent may be considered as same for all logical purposes. The main equivalences are listed below:

(*i*) **Commutative laws:** $(A \vee B) \Leftrightarrow (B \vee A), (A \wedge B) \Leftrightarrow (B \wedge A)$

(*ii*) **Associative laws:** $A \vee (B \vee C) \Leftrightarrow (A \vee B) \vee C$

$A \wedge (B \wedge C) \Leftrightarrow (A \wedge B) \wedge C$

(*iii*) **Distributive laws:** $A \wedge (B \vee C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$

$A \vee (B \wedge C) \Leftrightarrow (A \vee B) \wedge (A \vee C)$

(*iv*) **Idempotent laws:** $(A \vee A) \Leftrightarrow A, (A \wedge A) \Leftrightarrow A$

(*v*) **Absorption laws:** $A \vee (A \wedge B) \Leftrightarrow A, A \wedge (A \vee B) \Leftrightarrow A$

(*vi*) **De Morgan's laws:** $\neg(A \vee B) \Leftrightarrow (\neg A) \wedge (\neg B)$

$\neg(A \wedge B) \Leftrightarrow (\neg A) \vee (\neg B)$

(*vii*) **Detachment:** $[(A \Rightarrow B)] \wedge A) \Rightarrow B$

(*viii*) **Disjunctive addition:** $A \Rightarrow (A \vee B)$

(*ix*) **Conjuctive simplification:** $(A \wedge B) \Rightarrow A$

(*x*) **Chain rule:** $[(A \Rightarrow B) \wedge (B \Rightarrow C)] \Rightarrow (A \Rightarrow C)$

(*xi*) **Disjunctive simplification:** $[(A \vee B) \wedge (\neg A)] \Rightarrow B$

(*xii*) **Conditional rules:** $(A \Rightarrow B) \Leftrightarrow [(\neg A) \vee B]$

$(A \Rightarrow B) \Leftrightarrow \neg B \Rightarrow \neg A$

(*xiii*) **Equivalence rules:** $(A \Leftrightarrow B) \Leftrightarrow (A \Leftrightarrow B) \wedge (B \Rightarrow A)$

$(A \Leftrightarrow B) \Leftrightarrow (A \wedge B) \vee [(\neg A) \wedge (\neg B)]$

$[\neg (A \Rightarrow B)] \Leftrightarrow [A \Leftrightarrow (\neg B)]$

(*xiv*) **Implecation rule:** $(A \Rightarrow B) \Rightarrow (A \wedge C) \Rightarrow (B \wedge C)$

$(A \Rightarrow B) \Rightarrow (A \vee C) \Rightarrow (B \vee C)$

(*xv*) **Contraposition inference:** $[(A \Rightarrow B) \wedge (\neg B)] \Rightarrow (\neg A)$.

3.5 Well-Formed Formulae (WFF)

A statement formulae is not actually a statement. A statement can be obtained from it by replacing the variable by statements. The language consists of the following mathematical order (symbols):

(*i*) Connectives (negation, disjunction, conjunction *etc.*,).

(*ii*) Equality (=).

(*iii*) Elementhood or belongingness "*e*".

(*iv*) Parantheses (,).

(*v*) If P and Q are formulae, then $P \vee Q$ is a formula.

(*vi*) The only formulae are those given by the rules (*i*) – (*v*).

3.6 Tautologies

For the moment we may say that statement formulae are compound statements derived from some statement variables. Statement formulae are not statements, they become statement when the statements variables used to define them are replaced by definite statement. The truth values of the resulting statement depend upon the truth values of the statement substituted for the variables. Thus, the truth table of a resulting statement is the summary of all its truth values for all possible assignment of values to the variables appearing in the formula. Therefore sometimes the truth values of the resulting statement may be T and F. There are some formulae whose truth values always T or always F irrespective of the truth values assigned to the variables. A statement whose truth value is always T is

called a tautology and one whose truth value is always false is called a *contradiction*.

Note: The negation of a tautology is a contradiction and that of a contradiction is a tautology.

e.g., let P be any statement and consider the statements.

$P \vee \neg P$, and $P \wedge \neg P$. The truth tables are given below. Observe that $P \vee \neg P$ is a tautology and $P \wedge \neg P$ is a contradiction.

P	$\neg$P	P $\vee$ $\neg$P	P $\wedge$ $\neg$P
T	F	T	F
F	T	T	F

Tautology Implications

1. $p \wedge q \Rightarrow p$
2. $p \wedge q \Rightarrow q$
3. $p \Rightarrow p \vee q$
4. $\neg p \Rightarrow p \rightarrow q$
5. $q \Rightarrow p \rightarrow q$
6. $\neg(p \rightarrow q) \Rightarrow p$
7. $\neg(p \rightarrow q) \Rightarrow \neg p$
8. $p \wedge (p \rightarrow q) \Rightarrow q$
9. $\neg q \wedge (p \rightarrow q) \Rightarrow \neg p$
10. $\neg p \wedge (p \vee q) \Rightarrow q$
11. $(p \rightarrow q) \wedge (q \rightarrow r) \Rightarrow p \rightarrow r$
12. $(p \vee q) \wedge (p \rightarrow r) \wedge (q \rightarrow r) \Rightarrow r$

3.7 Principle of Duality

Let us consider two formulae A and A^*, these are said to be duals of each other if either one can be obtained from the other by replacing $\wedge$ by $\vee$ and $\vee$ by $\wedge$. There two connectives are termed as dual of each other. Let formula A contains the special variables T or F, then its dual A is obtained by replacing T by F and F by T in addition to the changes suggested above. There is no loss of generality if we confine only to the symbols $\vee$, $\wedge$ and $\neg$, because can say that formula containing any other connectives can be replaced by an equivalent formula containing only these three.

i.e., (*i*) The dual of $(P \vee Q) \wedge R$ is $(P \wedge Q) \vee R$

(*ii*) The dual of $(P \wedge Q) \vee T$ is $(P \vee Q) \wedge T$

(*iii*) The dual of $\neg(P \vee Q) \wedge (P \vee \neg(Q \wedge \neg S))$ is $\neg(P \wedge Q) \vee (P \wedge \neg(Q \vee \neg S))$

Duality Theorem

If $A(p_1, p_2, ..., p_n) \equiv B(p_1, p_2, ..., p_n)$, where A and B are compound propositons, then $A^*(p_1, p_2, ..., p_n) \equiv B^*(\boldsymbol{p}_1, p_2, ..., p_n)$.

Proof

We have

$$\neg(p \vee q) \equiv \neg p \wedge \neg q \quad \text{or} \quad p \vee q \equiv \neg(\neg p \wedge \neg q) \qquad ...(1)$$

Similarly we can prove that

$$p \wedge q \equiv \neg(\neg p \vee \neg q) \qquad ...(2)$$

(1) and (2) are known as De Morgan's laws.

Using (1) and (2), we can show that

$$\neg A(p_1, \boldsymbol{p}_2, ..., p_n) \equiv A^*(\neg p_1, \neg p_2, ..., \neg p_n) \qquad ...(3)$$

Equation (3) means that the negation of a proposition is equivalent to its dual in which every variable (primary proposition) is replaced by its negation. From Eq. (3), it follow that

$$A(p_1, p_2, ..., p_n) \equiv \neg A^*(\neg p_1, \neg p_2, ..., \neg p_n) \qquad ...(4)$$

Now since $A(p_1, p_2, ..., p_n) \equiv B(p_1, p_2, ..., p_n)$, we have $A(p_1, p_2, ..., p_n) \leftrightarrow B(p_1, p_2, ..., p_n)$ is tautology

$\therefore$ $A(\neg p_1, \neg p_2, ..., \neg p_n) \leftrightarrow B(\neg p_1, \neg p_2, ..., \neg p_n)$ is also a tautology ...(5)

Using (4) in (5), we get

$\neg A^*(p_1, p_2, ..., p_n) \leftrightarrow \neg B^*(p_1, p_2, ..., p_n)$ is a tautology.

$\therefore$ $A^* \leftrightarrow B^*$ is a tautology.

$\therefore$ $A^* \equiv B^*$. ■

Example 1

Construct truth table for the following formula.

$$[(\neg Q) \wedge (P \Rightarrow Q)] \Leftrightarrow P.$$

Solution

P	Q	⌝Q	P ⇒ Q	⌝Q ∧ (P ⇒ Q)	∨
T	T	F	T	F	F
T	F	T	F	F	F
F	T	F	T	F	T
F	F	T	T	T	F

Example 2

Establish the equivalence of the folowing:

(*i*) $P \Rightarrow (Q \Rightarrow R) \equiv (P \wedge Q) \Rightarrow R$

(*ii*) $(P \Rightarrow Q) \Rightarrow (P \wedge Q) \equiv (\neg P \Rightarrow Q) \wedge (Q \Rightarrow P)$

Solution

(*i*) $P \Rightarrow (Q \Rightarrow R) \equiv \neg P \vee (Q \Rightarrow R) \vee \neg P \equiv (\neg Q \vee R)$

$\equiv (\neg P \vee \neg Q) \vee R \equiv \neg(P \wedge Q) \vee R$

$\equiv (P \wedge Q) \Rightarrow R$

(*ii*) $(P \Rightarrow Q) \Rightarrow (P \wedge Q) \equiv \neg(P \Rightarrow Q) \vee (P \wedge Q)$

$\equiv (P \wedge \neg Q) \vee (P \wedge Q) \equiv P \wedge (Q \vee \neg Q)$

$(\neg P \Rightarrow Q) \wedge (Q \Rightarrow P) \equiv (P \vee Q) \wedge (\neg Q \vee P) \equiv (P \vee Q) \wedge (P \vee \neg Q)$

$\equiv P \vee (Q \wedge \neg Q)$

Now if $P \wedge (Q \vee \neg Q)$ is true, then so is P. Hence $P \vee (Q \wedge \neg m)$ is true. On the other hand if $P \vee (Q \wedge \neg Q)$ is true. Then at least one of then is true. But $Q \wedge \neg Q$ false always.

Therefore P is true. This $P \wedge (Q \vee \neg Q)$ is also true because $Q \vee \neg Q$ is true always. Thus these are true and identical cases and thus these are equivalent. ■

Example 3

Establish the equivalence of

$$(P \wedge Q) \vee [\neg P \vee (\neg P \vee Q)] \equiv \neg P \vee Q.$$

Also write its dual equivalence.

Solution

L.H.S. $= (P \wedge Q) \vee [\neg P \vee (\neg P \vee Q)]$

$= (P \wedge Q) \vee [(\neg P \vee \neg P) \vee Q)]$

$= (P \wedge Q) \vee (\neg P \vee Q)$

Now $(P \wedge Q) \wedge (\neg P \vee Q) = [(P \wedge Q) \wedge \neg P] \vee [(P \wedge Q) \wedge Q]$

$= [(P \wedge Q) \wedge \neg P] \vee (P \wedge Q) = (P \wedge Q)$ (by absorption)

$\therefore \quad (P \wedge Q) \vee (\neg P \vee Q) = [(P \wedge Q) \wedge (\neg P \vee Q)] \vee (\neg P \vee Q)$

$= \neg P \vee Q$ (by absorption)

= R.H.S.

The dual equivalence is

$(P \vee Q) \wedge [\neg P \wedge (\neg P \wedge Q)] = \neg P \wedge Q.$ ■

Example 4

Construct the truth table for

$$(P \Rightarrow Q) \wedge (Q \Rightarrow P)$$

Solution

P	Q	$P \to Q$	$Q \to P$	$(P \Rightarrow Q) \wedge (Q \Rightarrow P)$
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

Example 5

Construct the truth table for the formula.

$$\neg(P \wedge Q) \Leftrightarrow (\neg P \vee \neg Q)$$

Solution

P	Q	$P \wedge Q$	$\neg(P \wedge Q)$	$\neg P$	$\neg Q$	$\neg P \vee \neg Q$	$\neg(P \wedge Q) \Leftrightarrow (\neg P \vee \neg Q)$
T	T	T	F	F	F	F	T
T	F	F	T	F	T	T	T
F	T	F	T	T	F	T	T
F	F	F	T	T	T	T	T

Example 6

Show that $[\neg P \wedge (\neg Q \wedge R)] \vee (Q \wedge R) \vee (P \wedge R) \Leftrightarrow R.$

Solution

$$[\neg P \wedge (\neg Q \wedge R)] \vee (Q \wedge R) \vee (P \wedge R)$$

$\Leftrightarrow \quad [\neg P \wedge (\neg Q \wedge R)] \vee [(Q \vee P) \wedge R]$

$\Leftrightarrow \quad [(\neg P \wedge \neg Q) \wedge R] \vee [(Q \vee P) \wedge R]$

$\Leftrightarrow \quad [(\neg P \wedge \neg Q) \vee (Q \vee P)] \wedge R$

$\Leftrightarrow \quad [\neg (P \vee Q) \vee (P \vee Q)] \wedge R$

$\Leftrightarrow \quad T \wedge R \Leftrightarrow R.$ ■

Example 7

Show that

$[(P \vee Q) \wedge \neg \{\neg P \wedge (\neg Q \vee \neg R)\}] \vee (\neg P \wedge \neg Q) \vee (\neg P \wedge \neg R)$ *is a toutology.*

Solution

Using De-Morgan's Laws, we obtain

$$\neg P \wedge \neg Q \Leftrightarrow \neg (P \text{ d } Q) \quad \neg P \wedge \neg R \Leftrightarrow \neg (P \text{ d } R)$$

$$(\neg P \wedge \neg Q) \vee (\neg P \wedge \neg R) \Leftrightarrow \neg (P \vee Q) \vee \neg (P \vee R)$$

$$\Leftrightarrow \neg [(\mathrm{P} \vee \mathrm{Q}) \wedge (\mathrm{P} \vee \mathrm{R})]$$

Also $\neg [\neg P \wedge (\neg Q \vee \neg R)] \Leftrightarrow \neg [\neg P \wedge \neg (Q \wedge R)]$

$$\Leftrightarrow P \vee (Q \wedge R) \Leftrightarrow (P \vee Q) \wedge (P \vee R)$$

$(P \vee Q) \wedge (P \vee Q) \wedge (P \vee R) \Leftrightarrow (P \vee Q) \wedge (P \vee R)$

Consequently, the given formula is equivalent to

$[(P \vee Q) \wedge (P \vee R)] \vee \neg [(P \vee Q) \wedge (P \vee R)]$

Which is substitution instance of $P \vee \neg P$. ■

Example 8

Show that

$$\neg (P \wedge Q) \Rightarrow [\neg P \vee (\neg P \vee Q)] \Leftrightarrow (\neg P \vee Q)$$

Solution

$$\neg (P \wedge Q) \Rightarrow [\neg \mathrm{P} \vee (\neg \mathrm{P} \vee \mathrm{Q})]$$

$$\Leftrightarrow (P \wedge Q) \vee [\neg P \vee (\neg P \vee Q)]$$

$$\Leftrightarrow (P \wedge Q) \vee (\neg P \vee Q)$$

$$\Leftrightarrow (P \wedge Q) \vee \neg P \vee Q$$

$$\Leftrightarrow [(P \vee \neg P) \wedge (Q \vee \neg P)] \vee Q$$

$$\Leftrightarrow (Q \vee \neg P) \vee Q \Leftrightarrow Q \vee \neg P \Leftrightarrow \neg P \vee Q.$$

It follows that

$$P \wedge Q \vee [\neg P \vee (\neg P \vee Q)] \Leftrightarrow \neg P \vee Q.$$ ■

3.8 Two State Devices

The practical logic is called two-valued or bi-valued logic becouse in a framework any statement is allowed to have one and exactly one of the two possible truth values T (true) and F (False) which can also be written as 1 or 0 respectively.

There exist electrical and mechanical devices having one of the two possible configurations like 'on' and 'off'. Here 'on' will represent T or 1 and 'off' for F or 0. That's why these devices are so called **two state devices**.

For example, if we consider on electric lamp controlled by a mechanical switch. Let p denotes the input switch and s the output indicate lamp. When p is open, no current flow in the entire circuit and the lamp s is 'off'. When p is closed, s is 'on'. Let the statement be denoted by;

P: The switch p is closed .

S: The lamp s is on.

This could be expressed as a truth table or as a table for input/output values is the following manner;

State of p	State of s
Closed	On
Open	Off

or

p(P)	s(S)
1	1
0	0

We consider a circuit where there are two switches P and Q in a series. The lamp in this case is turned on whenever both the switches P and Q are closed.

Let P: The switch P is closed.

Q: The switch Q is closed.

S: The light s is on

Then, it will be clear that $P \wedge Q \Leftrightarrow S$.

P	Q	S
1	1	1
1	0	0
0	1	0
0	0	0

The circuit shows a two-state devices for AND logic:

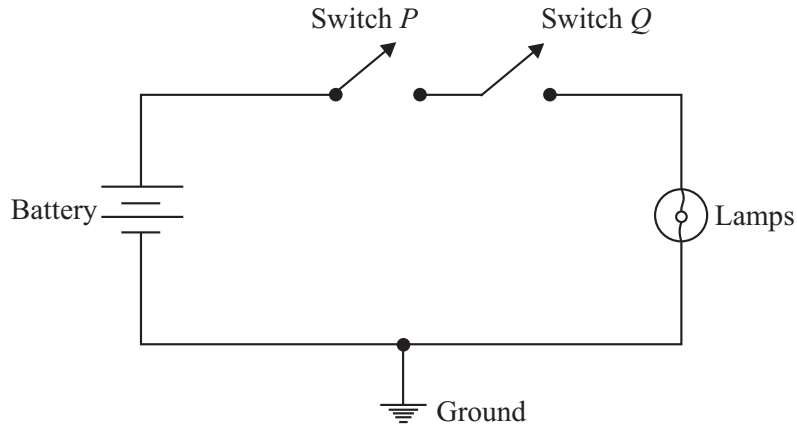


Fig. 3.1: Two state device circuit for AND logic

If the two switches are counected in parallel, then $P \vee Q \Leftrightarrow S$

P	Q	S
1	1	1
0	1	1
1	0	1
0	0	0

The following circuit shows a two state device for OR logic.

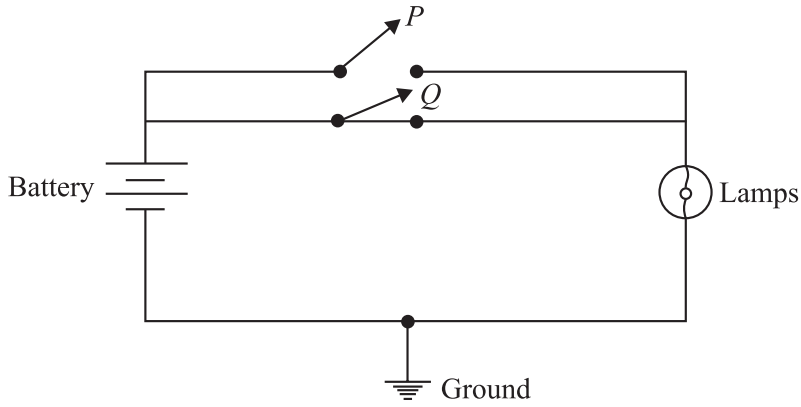


Fig. 3.2: Two state device circuit for OR logic

3.9 The Relay-Switching Devices

If a switch is controlled by a relay circuit, it is called a relay-switching device. On this device, when a switch P is open (P is false *i.e.* p: the switch p is closed), no current flows and the contact Q switch is normally

closed remains closed and the contact r remains open. When P is closed, the current will flow from the battery through the coil which will cause the movement of a relay armature, which in turn causes the springs to move downward and normally closed contact Q to open while the normally open contact r closes. If P is opened the contact q closes and r opens because the spring moves upward to its original position. With the statements P, Q and R denoting the switches p, q and r to be closed respectively, we can represent the operation of the device by the table of combinations (as shown is fig. 3.3). The switches q and r are always in the opposite state, *i.e.* $Q \Leftrightarrow \neg R$, $Q \Leftrightarrow \neg P$ and $R \Leftrightarrow P$.

Note: The output Q is the negation of the output P.

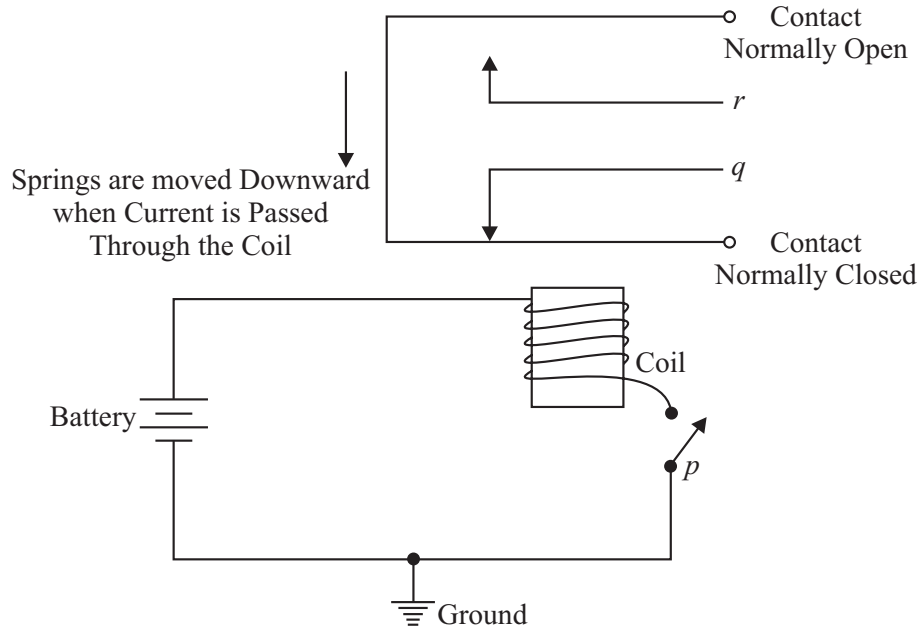


Fig. 3.3: A relay-switching device

P	Q	R
1	0	1
0	1	0

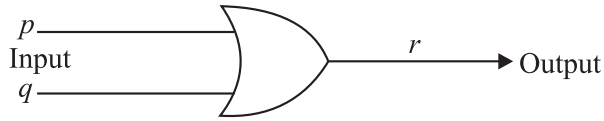
3.10 Logic Gates and Modules

3.10.1 OR, AND and NOT Gates

Instead of representing $\neg$, $\vee$, $\wedge$ by a circuit in the manner the circuits can also be represented by block diagrams, each of such diagram having one

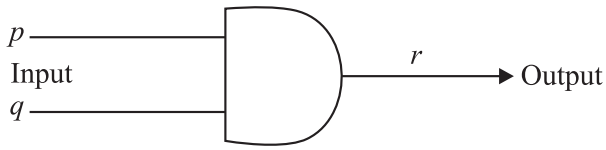
or many input wires and one output wire. Such diagramatic representation of logic circuits is called gates.

In gates, the connectives $\neg$, $\vee$ and $\wedge$ are usually denoted by the symbols '–' '+' and '.' respectively. The block diagrams and tables for OR gate. AND gate and NOT gate are given respectively in the following diagrams.



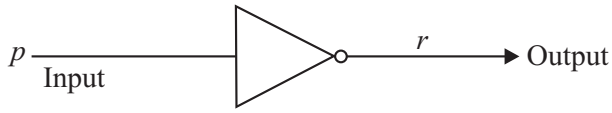
An OR-gate

p	q	r (p + q)
1	1	1
1	0	1
0	1	1
0	0	0



An AND-gate

p	q	r (p.q)
1	1	1
1	0	0
0	1	0
0	0	0



A NOT-gate

p	r(p)
1	0
0	1

Fig. 3.4: Symbolic gates and their Truth tables

3.10.2 Two-Level Networks

A two level network is defined as a network in which the longest path through which information passes from input to output is two gates. A two-level network consisting of AND gates at the input stage followed by OR-gates at the output stage is called a sum-of-product (AND-OR) network. An example of such a network is given by the expression $a.b + c.d$.

Another type of two-level network is to have OR-gates at the input stage and then AND-gates at the output stage and such an expression is of the form $(a + b).(c + d)$. This is called product-of-sum (OR-AND) network as shown below.

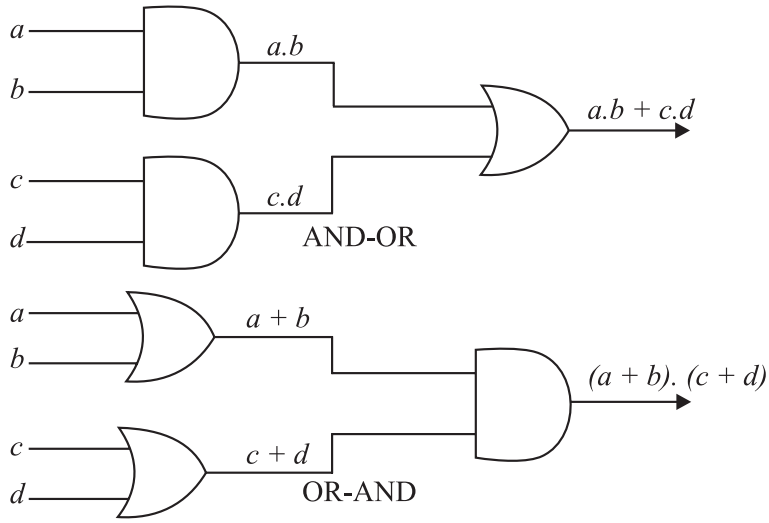


Fig. 3.5: Combination of two level networks

3.10.3 NOR and NAND Gates

There exists a combination of two logic gates is practice. There are two other types of gates which are useful and frequently used in computer science. These are called NOR and NAND gates *i.e.* NOR (NOT + OR), NAND (NOT + AND). The following tables show each one of them, their tables and the equivalent representations.

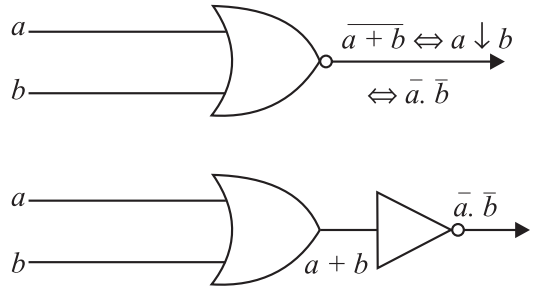


Fig. 3.6 NOR and NAND Gates

Input		Output
a	b	
1	1	0
1	0	0
0	1	0
0	0	1

Fig. 3.7 Truth table for NOR Gate

Input		Output
a	b	
1	1	0
1	0	1
0	1	1
0	0	1

Fig. 3.7: *(a)* Truth table for NAND Gate

Solved Examples (Based as symbolic logic)

Example 1

Express $P \Rightarrow Q$ and $P \Leftrightarrow Q$ by using $\uparrow$ and $\downarrow$ only.

Solution

$$P \Rightarrow Q \equiv \urcorner P \vee Q$$
$$= \urcorner P \wedge \urcorner (\urcorner Q)$$
$$= P \uparrow \urcorner Q.$$
$$= P \uparrow (Q \uparrow Q)$$

$$P \Leftrightarrow Q = (P \Rightarrow Q) \wedge (Q \Rightarrow P)$$
$$= \neg[\, P \Rightarrow Q) \uparrow (Q \Rightarrow P)]$$
$$= [(P \Rightarrow Q) \uparrow (Q \Rightarrow P)] \uparrow [(P \Rightarrow Q) \uparrow (\mathrm{Q} \Rightarrow \mathrm{P})]$$
$$= [\{P \uparrow (Q \uparrow Q)\} \uparrow \{Q \uparrow (P \uparrow P)\}] \uparrow [\{P \uparrow (Q \uparrow Q)\} \uparrow \{Q \uparrow (P \uparrow P)\}]$$

These expressions are not unique, another such expression would be.

$$P \Rightarrow Q = (\neg P \vee Q) \wedge (P \vee \neg Q)$$
$$= [(\neg P \vee Q) \wedge P] \vee [(\neg P \vee Q) \wedge \neg Q]$$
$$= (\neg P \wedge P) \vee (Q \wedge P) \vee (\neg P \vee \neg Q) \vee (Q \wedge \neg Q)$$
$$= (P \wedge Q) \vee (\neg P \wedge \neg Q)$$
$$= (P \wedge Q) \vee (P \vee Q)$$
$$= \neg\neg((P \wedge Q) \vee \neg(P \vee Q)]$$
$$= \neg[\neg(P \wedge Q) \wedge (P \vee Q)]$$
$$= \neg(P \wedge Q) \uparrow (P \vee Q)$$
$$= (P \uparrow Q) \uparrow [(P \uparrow P) \uparrow (Q \uparrow Q)]$$ ■

Example 2

Express $P \downarrow Q$ by using $\uparrow$ only.

Solution

We have
$$P \downarrow Q \equiv \neg(P \vee Q)$$
$$\equiv (P \vee Q) \uparrow (P \vee Q)$$
$$\equiv [(P \uparrow P) \uparrow (Q \uparrow Q)] \uparrow [(P \uparrow P) \uparrow (Q \uparrow Q)]$$ ■

Example 3

Establish De-Morgan's Law by using $\uparrow$ and $\downarrow$.

i.e. $\neg(P \uparrow Q) \equiv \neg P \downarrow \neg Q.$

Solution

$$\text{R.H.S} \equiv \neg P \downarrow \neg Q$$
$$\equiv \neg(\neg P \vee \neg Q) = P \wedge Q.$$
$$\equiv \neg(\neg(P \wedge Q))$$
$$\equiv \neg(P \uparrow Q) = \text{L.H.S.}$$ ■

Example 4

Construct a truth table for each of the following compound propositions:

(*a*) $(p \vee q) \rightarrow (p \wedge q)$;

(*b*) $(p \rightarrow q) \rightarrow (q \rightarrow p)$;

(*c*) $(q \rightarrow \neg p) \leftrightarrow (p \leftrightarrow q)$;

(*d*) $(p \leftrightarrow q) \leftrightarrow ((p \wedge q) \vee (\neg p \wedge \neg q))$;

(*e*) $(\neg p \leftrightarrow \neg q) \leftrightarrow (p \leftrightarrow q)$.

Solution

(*a*) **Truth Table for $(p \vee q) \rightarrow (p \wedge q)$**

p	q	$p \vee q$	$p \wedge q$	$(p \vee q) \rightarrow (p \wedge q)$
T	T	T	T	T
T	F	T	F	F
F	T	T	F	F
F	F	F	F	T

(*b*) **Truth Table for $(p \rightarrow q) \rightarrow (q \rightarrow p)$**

p	q	$p \rightarrow q$	$q \rightarrow p$	$(p \rightarrow q) \rightarrow (q \rightarrow p)$
T	T	T	T	T
T	F	F	T	T
F	T	T	F	F
F	F	T	T	T

(*c*) **Truth Table for $(q \rightarrow \neg p) \leftrightarrow (q \leftrightarrow p)$**

p	q	$\neg p$	$q \rightarrow \neg q$	$p \leftrightarrow q$	$(q \rightarrow \neg p) \leftrightarrow (p \leftrightarrow q)$
T	T	F	F	T	F
T	F	F	T	F	F
F	T	T	T	F	F
F	F	T	T	T	T

(*d*) Truth Table for $(p \leftrightarrow q) \leftrightarrow ((p \wedge q) \vee (\neg p \wedge \neg q))$

p	q	¬p	¬q	p ↔ q	p ∧ q	¬p ∧ ¬q	(p ∧q) ∨ (¬p ∧ ¬q)	given formula
T	T	F	F	T	T	F	T	T
T	F	F	T	F	F	F	F	T
F	T	T	F	F	F	F	F	T
F	F	T	T	T	F	T	T	T

(*e*) Truth Table for $(\neg p \vee \neg q) \leftrightarrow (p \leftrightarrow q)$

p	q	¬p	¬q	(¬p ↔ ¬q)	(p ↔ q)	(¬p ↔ ¬q) ↔ (p ↔ q)
T	T	F	F	T	T	T
T	F	F	T	F	F	T
F	T	T	F	F	F	T
F	F	T	T	T	T	T

Example 5

Construct the truth table for each of the compound propositions given as follows:

(*a*) $((p \rightarrow (q \rightarrow r)) \rightarrow ((p \rightarrow q) \rightarrow (p \rightarrow r))$
(*b*) $\neg(p \vee (q \wedge r)) \leftrightarrow ((p \vee q) \wedge (p \rightarrow r))$
(*c*) $(\neg p \leftrightarrow \neg q) \leftrightarrow (q \leftrightarrow r)$
(*d*) $(p \rightarrow (q \rightarrow s)) \wedge (\neg r \vee p) \wedge q$
(*e*) $((p \rightarrow q) \rightarrow r) \rightarrow s$

Solution

(*a*) Truth Table for $(p \rightarrow (q \rightarrow r)) \rightarrow ((p \rightarrow q) \rightarrow (p \rightarrow r))$

p	q	r	p → q	p → r	q → r	p → (q → r) ≡ a	(p → q) → (p → r) ≡ b	a → b
T	T	T	T	T	T	T	T	T
T	T	F	T	F	F	F	F	T
T	F	T	F	T	T	T	T	T
T	F	F	F	F	T	T	T	T
F	T	T	T	T	T	T	T	T
F	T	F	T	T	F	T	T	T
F	F	T	T	T	T	T	T	T
F	F	F	T	T	T	T	T	T

(*b*) Truth Table for $\neg(p \vee (q \wedge r) \leftrightarrow ((p \vee q) \wedge (p \rightarrow r)$

p	q	r	$q \wedge r$	$p \vee (q \wedge r) \equiv a$	$\neg a$	$p \vee q$	$p \rightarrow r$	$(p \vee q) \wedge (p \rightarrow r) \equiv b$	$\neg a \leftrightarrow b$
T	T	T	T	T	F	T	T	T	F
T	T	F	F	T	F	T	F	F	T
T	F	T	F	T	F	T	T	T	F
T	F	F	F	T	F	T	F	F	T
F	T	T	T	T	F	T	T	T	F
F	T	F	F	F	T	T	T	T	T
F	F	T	F	F	T	F	T	F	F
F	F	F	F	F	T	F	T	F	F

(*c*) Truth Table for $(\neg p \leftrightarrow \neg q) \leftrightarrow (q \leftrightarrow r)$

p	q	r	$\neg p$	$\neg q$	$(\neg p \leftrightarrow \neg q) \equiv a$	$q \leftrightarrow r \equiv b$	$a \leftrightarrow b$
T	T	T	F	F	T	T	T
T	T	F	F	F	T	F	F
T	F	T	F	T	F	F	T
T	F	F	F	T	F	T	F
F	T	T	T	F	F	T	F
F	T	F	T	F	F	F	T
F	F	T	T	T	T	F	F
F	F	F	T	T	T	T	T

(*d*) Truth Table for $(p \rightarrow (q \rightarrow s)) \wedge (\neg r \vee p) \wedge q$

p	q	r	s	$q \rightarrow s \equiv a$	$p \rightarrow a \equiv b$	$\neg r$	$(\neg r \vee p) \equiv c$	$b \wedge c$	$b \wedge c \wedge q$
T	T	T	T	T	T	F	T	T	T
T	T	T	F	F	F	F	T	F	F
T	T	F	T	T	T	T	T	T	T
T	T	F	F	F	F	T	T	F	F
T	F	T	T	T	T	F	T	T	F
T	F	T	F	T	T	F	T	T	F
T	F	F	T	T	T	T	T	T	F
T	F	F	F	T	T	T	T	T	F
F	T	T	T	T	T	F	F	F	F

F	T	T	F	F	T	F	F	F	F
F	T	F	T	T	T	T	T	T	T
F	T	F	F	F	T	T	T	T	T
F	F	T	T	T	T	F	F	F	F
F	F	T	F	T	T	F	F	F	F
F	F	F	T	T	T	T	T	T	F
F	F	F	F	T	T	T	T	T	F

(*e*) Truth Table for ((p → q) → r) → s

p	q	r	s	p → q	(p → q) → r	((p → q) → r) → s
T	T	T	T	T	T	T
T	T	T	F	T	T	F
T	T	F	T	T	F	T
T	T	F	F	T	F	T
T	F	T	T	F	T	T
T	F	T	F	F	T	F
T	F	F	T	F	T	T
T	F	F	F	F	T	F
F	T	T	T	T	T	T
F	T	T	F	T	T	F
F	T	F	T	T	F	T
F	T	F	F	T	F	T
F	F	T	T	T	T	T
F	F	T	F	T	T	F
F	F	F	T	T	F	T
F	F	F	F	T	F	T

Example 6

Without using truth tables, prove the following:

(a) $(\neg p \vee q) \wedge (p \wedge (p \wedge q) \equiv p \wedge q$

(b) $p \rightarrow (q \rightarrow p) \equiv \neg p \rightarrow (p \rightarrow q)$

(c) $\neg(p \leftrightarrow q) \equiv (p \vee q) \wedge \neg (p \vee q) \equiv (p \wedge \neg q) \vee (\neg p \wedge q)$

Solution

$(a)\ (\neg p \vee q) \wedge (p \wedge (p \wedge q)) \equiv (\neg p \vee q) \wedge (p \wedge p) \wedge q$, (by associative law)

$\equiv (\neg p \vee q) \wedge q) \wedge (p \wedge q)$, (by idempotent law)

$\equiv (p \wedge q) \wedge (\neg p \vee q)$, (by commutative law)

$\equiv ((p \wedge q) \wedge \neg p) \vee ((p \wedge q) \wedge q)$, (by distributive law)

$\equiv (\neg p \wedge (p \wedge q)) \vee ((p \wedge q) \wedge q)$, (by commutative law)

$\equiv ((\neg p \wedge p) \wedge q) \vee (p \wedge (q \wedge q)$, (by associative law)

$\equiv (F \vee q) \vee (p \wedge q)$, (by complement and idempotent law)

$\equiv F \vee (p \wedge q)$, (by dominant law)

$\equiv p \wedge q$, (by dominant law).

(*b*) $p \to (q \to p) \equiv \neg p \vee (q \to p)$

$\equiv \neg p \vee (\neg q \vee p)$

$\equiv \neg q \vee (p \vee \neg p)$, (by commutative and associative laws)

$\equiv \neg p \vee \mathrm{T}$, (by complement law)

$\equiv T$, (by dominant law) ...(1))

$\neg p \to (p \to q) \equiv p \vee (p \to q)$

$\equiv p \vee (\neg p \vee q)$

$\equiv (p \vee \neg p) \vee q$, (by associative law)

$\equiv T \vee q$, (by complement law)

$\equiv T$, (by dominant law,) ...(2)

From (1) and (2), the result follows.

(*c*) $\neg (p \leftrightarrow q) \equiv \neg((p \to q) \wedge (q \to p))$

$\equiv \neg ((\neg p \vee q) \wedge (\neg q \vee q))$

$\equiv \neg[((\neg p \vee q) \wedge (\neg q) \vee ((\neg p \vee q) \wedge p]$, (by distributive law)

$\equiv \neg[((\neg p \wedge \neg q) \vee (q \wedge \neg q)) \vee ((\neg p \wedge q)) \vee (q \wedge p))]$,

(by distributive law)

$\equiv \neg[((\neg p \wedge \neg q) \vee F) \vee ((F \vee (q \wedge p))]$, (by complement law)

$\equiv \neg[(\neg p + \neg q) \vee (q \wedge p)]$, (by identity law)

$\equiv \neg[\neg(p \vee q) \vee (q \wedge p)]$, (by De Morgan's law)

$\equiv (p \vee q) \wedge \neg(q \wedge p)$, (by De Morgan's law) ...(1)

$\equiv (p \vee q) \wedge (\neg q \vee \neg p)$, (by De Morgan's law)

$\equiv ((p \vee q) \wedge \neg q)) \vee ((p \vee q) \wedge \neg p))$, (by distributive law)

$\equiv ((p \wedge \neg q) \vee (q \wedge \neg q)) \vee ((p \wedge \neg p) \vee (q \wedge \neg p))$,

(by distributive law)

$\equiv ((p \wedge \neg q) \vee F) \vee ((F \vee (q \wedge \neg p))$, (by complement law)

$\equiv (p \wedge \neg q) \vee (q \wedge \neg p)$, (by identity law)

$\equiv (p \wedge \neg q) \vee (\neg p \wedge q)$, (by commutative law) ...(2)

From (1) and (2), the result follows. ■

Example 7

Prove the following equivalences by proving the equivalences of the duals:

(a) $\neg((\neg p \wedge q) \vee (\neg p \wedge \neg q)) \vee (p \wedge q) \equiv p$

(b) $(p \vee q) \rightarrow r \equiv (p \rightarrow r) \wedge (q \wedge r)$

(c) $(p \wedge (p \leftrightarrow q)) \rightarrow q \equiv T$

Solution

(*a*) The dual of the given equivalence is

$\neg((\neg p \vee q) \wedge (\neg p \vee \neg q)) \wedge (p \vee q) \equiv p$

Let us now prove the dual equivalence.

L.H.S. $\equiv \neg(\neg p \vee (q \wedge \neg q)) \wedge (p \vee q)$, (by distribution law)

$\equiv \neg(\neg p \vee F) \wedge (p \vee q)$, (by complement law)

$\equiv \neg(\neg p) \wedge (p \vee q)$, (by identity law)

$\equiv p \wedge (p \vee q)$

$\equiv p$, (by absorption law)

(*b*) $(p \vee q) \rightarrow r \equiv (p \rightarrow r) \wedge (q \rightarrow r)$

i.e., $\neg(p \vee q) \vee r \equiv (\neg p \vee r) \wedge (\neg q \vee r)$

Dual of this equivalence is

$\neg(p \wedge q) \wedge r \equiv (\neg p \wedge r) \vee (\neg q \wedge r)$

L.H.S. $\equiv (\neg p \vee \neg q) \wedge r$, (by De Morgan's law)

$\equiv (\neg p \wedge r) \vee (\neg q \wedge r)$, (by distributive law)

$\equiv$ R.H.S.

(*c*) $(p \wedge (p \leftrightarrow q)) \rightarrow q \equiv T$

i.e., $p \wedge ((p \rightarrow q) \wedge (q \rightarrow p)) \rightarrow q \equiv T$,

i.e., $p \vee ((\neg p \vee q) \wedge (\neg q \vee p)) \rightarrow q \equiv T$

i.e., $\neg(p \wedge ((\neg p \vee q) \wedge (\neg q \vee p))) \vee q \equiv T$

Dual of this equivalence is

$\neg(p \vee ((\neg p \wedge q) \vee (\neg q \wedge p))) \wedge q \equiv F$

L.H.S. $\equiv \neg[(p \wedge (\neg p \wedge q)) \vee (\neg q \wedge p)] \wedge q$, (by associative law)

$\equiv \neg[(T \wedge (p \vee q)) \vee (\neg q \wedge p)] \wedge q$,

(by distributive and complement laws)

$\equiv \neg[(p \wedge q) \vee (\neg q \wedge p)] \wedge q$, (by identity law)

$\equiv \neg[((p \vee q) \vee \neg q) \wedge ((p \vee q) \vee p))] \wedge q$, (by distributive law)

$\equiv \neg[p \vee T) \wedge (q \vee q)] \wedge q$,

(by idempotent and complement laws.)

$\equiv \neg[T \wedge (p \vee q)] \wedge q$, (by dominant law)

$\equiv \neg[p \vee q] \wedge q$, (by identity law)

$\equiv (\neg p \wedge \neg q) \wedge q$, (by De Morgan's law)

$\equiv (\neg p \wedge F)$, (by complement law)

$\equiv F$, (by dominant law). ■

3.11 Normal Forms (Decision Problems)

We consider a statement formula $A(P_1, P_2, P_3, ..., P_n)$ where $P_1, P_2, P_3, ..., P_n$ be atomic variables. If we assume all possible assignments of the truth values to $P_1, P_2, P_3, ..., P_n$ and obtain the resulting truth values of the formula A, then we get the truth table for A. Such a truth table contains $2n$ rows. The formula may have the truth value. ¬ for all possible assignments of the truth values to the variables $P_1, P_2, P_3, ..., P_n$. In this case, A is called identically true or *tautology.* If A has the truth value F for all possible assignments of the truth values to $P_1, P_2, ..., P_n$, then A is called identically false or a *contradiction.* Finally, if A has the truth valve ¬ for atleast on combination of truth values assigned to $P_1, P_2, P_3, ..., P_n$. Then A is called *satisfiable.* The problem of determining, a finite number of steps, whether a given statement formula is a tautology or a contradict ion or at learst statistical is called a *decision problem.*

The construction of truth table involves a finite number of steps, and, such as a decrision problem in the statement calculus always has a solution. Similarly, decision problem can be poor for other logical systems, particularly, for the predicate calculus.

3.11.1 Disjunctive Normal Form (*DNF*)

In this form the word "*product*" can be used in place of "conjunction" and "*sum*" in place of "disjunction" in our current discussion.

A product of the variables and their negation in a particular formula is called an elementary product. Similarly, *a sum of the variables and their negations is termed as an elementary sum.*

Let P and Q be any two atomic variables. Then P, $\neg P \wedge Q$, $\neg Q \wedge P \wedge \neg P$, $P \wedge \neg P$, and $Q \wedge \neg P$, are some examples of elementary products. On the order hand, P, $\neg P \vee Q$, $\neg Q \vee P \vee \neg P$, $P \vee \neg P$, and $Q \vee \neg P$, are examples of elementary sums of the two variables. Any part of an elementary sum or product which is itself an elementary sum or product is called a factor of the original elementary sum or product. Thus $\neg Q$, $P \wedge \neg P$, and $\neg Q \wedge P$, are some of the factors of $\neg Q \wedge P \wedge \neg P$. The following statements hold for elementary sums and products.

(*i*) A necessary and sufficient condition for an elementary product to be identically false is that it contains at least one pair of factors in which one is the negation of the other.

(*ii*) A necessary and sufficient condition for an elementary sum to be identically true is that it contains at least one pair of factors in which one is the negation of the other.

A formula which is equivalent to a given formula consisting of sum of elementary product is called a *disjunction normal form* of the given formula.

Note: If the connectives $\Rightarrow$ and $\Leftrightarrow$ appear in the given formula, then an equivalent formula can be obtained in which "$\Rightarrow$" and "$\Leftrightarrow$" are replaced by $\wedge$. $\vee$, and $\neg$. This statement would be true of any other connective yet undefined. The truth of this statement will become apparent after our discussion of principal disjunctive normal forms. Therefore, there is no loss of generality in assuming that the given formula contains the connectives $\wedge$, $\vee$, and $\neg$ only.

SOLVED EXAMPLES

Example 1

Obtain the disjunctive normal forms of

(a) $P \wedge (P \Rightarrow Q)$, *(b)* $\neg(P \vee Q) \Leftrightarrow (P \wedge Q)$

Solution

(*a*) $P \wedge (P \Rightarrow Q) \Leftrightarrow P \wedge (\neg P \vee Q)$,

$\Leftrightarrow (P \wedge \neg P) \vee (P \wedge Q)$

(*b*) $\neg(P \vee Q) \Leftrightarrow (P \wedge Q)$

$\Leftrightarrow (\neg(P \vee Q) \wedge (P \wedge Q)) \vee ((P \vee Q) \vee \neg(P \wedge Q))$.

[Using "$R \Leftrightarrow S$" $\Leftrightarrow (R \wedge S) \vee (\neg R \wedge \neg S)$]

$\Leftrightarrow (\neg P \wedge \neg Q) \wedge (P \wedge Q) \vee ((P \vee Q) \wedge (\neg P \vee \neg Q))$

$\Leftrightarrow ((\neg P \wedge \neg Q \wedge P \wedge Q) \vee (P \vee Q) \wedge (\neg P) \wedge (\neg Q))$

$$\Leftrightarrow (\neg P \wedge \neg Q \wedge P \wedge Q) \vee (P \wedge \neg P) \vee (Q \wedge \neg P) \vee (P \wedge \neg Q) \vee (Q \wedge \neg Q)) \quad \blacksquare$$

Example 2

Find the disjunctive normal forms of the following statements.

(a) $\neg(\neg(p \leftrightarrow q) \wedge r)$
(b) $p \vee (\neg p \rightarrow (q \vee (q \rightarrow \neg r)))$
(c) $p \wedge \neg(q \wedge r) \vee (p \rightarrow q)$
(d) $(p \wedge \neg(q \vee r)) \vee (((p \wedge q) \vee \neg r) \wedge p)$

Solution

$(a)\ \neg(\neg(p \leftrightarrow q) \wedge r) \equiv \neg(\neg((p \wedge q) \vee \neg p \wedge \neg p)) \wedge r)$
$\equiv \neg[(\neg(p \wedge q) \wedge \neg(\neg p \wedge \neg q)) \wedge r]$
$\equiv \neg[((\neg p \vee \neg q) \wedge (p \vee q)) \wedge r]$
$\equiv \neg[((\neg p \wedge \neg q) \vee \neg(\neg p \wedge \neg q)) \vee (\neg q \wedge p) \vee (\neg q \wedge q) \wedge r]$
(by extended distributive law)
$\equiv \neg[((\neg p \wedge q) \vee (\neg q \wedge p)) \wedge r]$
$\equiv \neg[((\neg p \vee \neg q) \wedge (\neg p \vee q) \wedge (q \vee \neg q) \wedge (q \vee p)) \wedge r]$
$\equiv \neg[((p \vee q) \wedge (\neg p \vee \neg q)) \wedge r]$
$\equiv \neg(p \vee q) \vee \neg(\neg p \vee \neg q) \vee \neg r$
$\equiv (\neg p \wedge \neg q) \vee (p \wedge q) \vee \neg r$

$(b)\ p \vee (\neg p \rightarrow (q \vee (q \rightarrow \neg r)))$
$\equiv p \vee (\neg p \rightarrow (q \vee (\neg q \vee \neg r)))$
$\equiv p \vee (p \vee (q \vee (\neg q \vee \neg r)))$
$\equiv p \vee p \vee q \vee \neg q \vee \neg r$
$\equiv p \vee q \vee \neg q \vee \neg r$

Note: The given statement is a tautology, as $p \vee (q \vee \neg q) \vee \neg r \equiv P \vee T \vee \neg r \equiv T$

$(c)\ p \wedge \neg(q \vee r) \wedge (p \rightarrow q)$
$\equiv p \wedge \neg(q \wedge r) \vee (\neg p \wedge q)$
$\equiv (p \wedge (\neg q \vee \neg r)) \vee (\neg p \vee q)$
$\equiv (p \wedge \neg q) \vee (p \wedge \neg r) \vee (\neg p \vee q)$
$\equiv (p \wedge \neg q) \vee (p \wedge \neg r) \vee \neg p \vee q$

$(d)\ (p \wedge \neg(q \vee r)) \vee (((p \wedge q) \vee \neg r) \wedge p)$
$\equiv (p \wedge (\neg q \wedge \neg r)) \vee ((p \wedge q) \wedge q) \vee (\neg r + p)$
$\equiv (p \wedge \neg q \wedge \neg r) \vee (p \wedge q) \vee (p \vee \neg r).$ $\blacksquare$

Example 3

Without constructing the trutt tables, find the principal disjunctive normal forms of the following statements:

(a) $(\neg p \to q) \wedge (q \leftrightarrow p)$
(b) $(p \wedge q) \vee (\neg p \wedge q) \vee (q \wedge r)$
(c) $p \wedge \neg(q \wedge r) \vee (p \to q)$
(d) $(q \vee (p \wedge r)) \wedge \neg((p \vee r) \wedge q)$

Solution

$$(a)\ (\neg p \to q) \wedge (q \leftrightarrow p) \equiv (p \vee q) \wedge ((q \wedge p) \vee (\neg q \wedge \neg p))$$
$$\equiv (p \vee q) \wedge ((p \wedge q) \vee \neg(p \vee q))$$
$$\equiv ((p \vee q) \wedge (p \wedge q)) \vee ((p \vee q) \wedge \neg(p \vee q))$$
$$\equiv ((p \vee q) \wedge (p \wedge q)) \vee F$$
$$\equiv (p \wedge (p \wedge q)) \vee ((q \wedge (p \wedge q))$$
$$\equiv (p \wedge q) \vee (p \wedge q)$$
$$\equiv p \wedge q$$

$$(b)\ (p \wedge q) \vee (\neg p \wedge q) \vee (q \wedge r)$$
$$\equiv ((p \wedge q) \wedge (r \vee \neg r)) \vee ((\neg p \wedge q) \wedge (r \vee \neg r)) \vee ((q \wedge r) \wedge (p \vee \neg p))$$

($\because$ Already the given statement is in the DNF)

$$\equiv (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (p \wedge q \wedge r) \vee (\neg p \wedge q \wedge r)$$
$$\equiv (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r)$$

(Deleting repetition of identical minterms)

$$(c)\ p \wedge \neg(q \wedge r) \vee (p \to q)$$
$$\equiv (p \wedge (\neg q \vee \neg r)) \vee (\neg p \vee q)$$
$$\equiv (p \wedge \neg q) \vee (p \wedge \neg r) \vee (\neg p \vee q)$$
$$\equiv (p \wedge \neg q) \vee (p \wedge \neg r) \vee (\neg p \wedge (q \vee \neg q)) \vee (q \wedge (p \vee \neg p))$$
$$\equiv (p \wedge \neg q) \vee (p \wedge \neg r) \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q) \vee (p \wedge q)$$

[Omitting the repletion of $(\neg p \wedge q)$]

$$\equiv ((p \wedge \neg q) \wedge (r \vee \neg r)) \vee ((p \wedge \neg r) \wedge (q \vee \neg q)) \vee ((\neg p \wedge q) \wedge (r \vee \neg r) \wedge ((\neg p \wedge \neg q) \wedge (r \vee \neg r)) \vee ((p \wedge q) \vee ((r \vee \neg r))$$

$$\equiv (p \wedge \neg q \wedge r) \vee (p \wedge \neg q \wedge \neg r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge \neg r)$$
$$\vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r)$$
$$\vee (\neg p \wedge \neg q \vee \neg r) \vee (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r)$$
$$\equiv (p \wedge \neg q \wedge r) \vee (p \vee \neg q \wedge \neg r) \vee (p \wedge q \wedge \neg r) \vee (\neg p \wedge q \wedge r)$$
$$\vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge \neg q \wedge \neg r) \vee (p \wedge q \wedge r)$$

(Omitting repetitions)

Note: Since all possible minterms are present in the PDNF, we infer that the given statement is a tautology.

(*d*) $(q \vee (p \wedge r)) \wedge \neg((p \vee r) \wedge q)$

$\equiv (q \vee (p \wedge r)) \wedge (\neg(p \vee r) \vee \neg q)$

$\equiv (q \vee (p \wedge r)) \wedge ((\neg p \wedge \neg r) \vee \neg q)$

$\equiv (q \wedge \neg p \wedge \neg r) \vee (q \wedge \neg q) \vee (p \wedge r \wedge \neg p \wedge \neg r) \vee (p \wedge r \wedge \neg q)$

(By extended distribution law)

$\equiv (\neg p \wedge q \wedge \neg r) \vee F \vee F \vee (p \wedge \neg q \wedge r)$

$\equiv (\neg p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge r)$, (deleting F's) ■

3.11.2 Conjunctive Normal Form (CNF)

A formula which is equivalent to a given formula and consisting of a product of elementary sums is termed as **conjunctive normal form** of the given formula.

The method of obtaining a CMF of a given formula is similar to the one giving for DNF and would be demonstrated through examples.

Note: (*i*) The CNF is not unique

(*ii*) A given formula is identically true if every elementary sum in its CNF is identically true.

Solved Examples

Obtain the CNF of the following formulae.

(*a*) $P \wedge (P \Rightarrow Q)$

(*b*) $\neg(P \vee Q) \Leftrightarrow P \wedge Q)$

Solution

(*a*) $P \wedge (P \Rightarrow Q)$

$\Leftrightarrow P \wedge (\neg P \vee Q)$

Hence $P \wedge (\neg P \vee Q)$ is the required form.

(*b*) $\neg(P \vee Q) \Leftrightarrow (P \wedge Q) \Leftrightarrow (\neg(P \vee Q) \rightarrow (P \vee Q)$

$\wedge ((P \wedge Q) \rightarrow \neg(P \vee Q))$

[Using $R \Leftrightarrow S \Leftrightarrow (R \rightarrow S) \wedge (S \rightarrow R)$]

$$\Leftrightarrow ((P \vee Q) \vee (P \wedge Q)) \wedge (\neg(P \wedge Q) \vee (\neg P \wedge \neg Q))$$
$$\Leftrightarrow ((P \vee Q \vee P) \wedge (P \wedge Q \vee Q)) \wedge ((\neg P \vee \neg Q) \vee (\neg P \wedge \neg Q)$$
$$\Leftrightarrow ((P \vee Q \vee P) \wedge (P \vee Q \vee Q)) \wedge ((\neg P \vee \neg Q) \vee (\neg P \wedge \neg Q))$$
$$\Leftrightarrow (P \vee Q \vee P) \wedge (P \vee Q \vee Q) \wedge (\neg P \vee \neg Q \vee \neg P) \vee (\neg P \vee \neg Q \vee \neg Q)$$

is the required form. ■

Example 2

Show that the formula $Q \vee (P \wedge \neg Q) \vee (\neg P \wedge \neg Q)$ is a tautology.

Solution

Firstly we get a CNF of the given formula.

$$Q \vee (P \wedge \neg Q) \vee (\neg P \wedge \neg Q) \Leftrightarrow Q \vee (P \vee \neg P) \wedge \neg Q]$$
$$\Leftrightarrow (Q \vee (P \wedge \neg P) \wedge (Q \vee \neg Q)$$
$$\Leftrightarrow (Q \vee P \vee \neg P) \wedge (Q \vee \neg Q)$$

Since each of the elementary sums is a tautology, the given formula is also a tautology. ■

3.11.3 Principal Disjunctive Normal Form (PDNF)

Let P and Q be two statement variables. Let us construct all possible formulas which consist of conjunctions of P or its negation and conjunctions of Q or its negation. None of the formulas should contain both a variable and its negation. Furthermore, any formula which is obtained by commuting the formulas in the conjunction is not included in the list because such a formula will be equivalent to one included in the list. For example, either $P \wedge Q$ or $Q \wedge P$ is included, but not both, For two variables P and Q, there are 2^2 such formulas given by

$$P \wedge Q \qquad P \wedge \neg Q \qquad \neg P \wedge Q \quad \text{and} \quad \neg P \wedge \neg Q$$

These formulas are called *minterms* or Boolean conjunctions of P and Q. From the truth tables of these minterms, it is clear that no two minterms are equivalent. Each minterm has the truth value T for exactly one combination of the truth values of the variables P and Q. This fact is shown in table below:

P	Q	$P \wedge Q$	$P \wedge \neg Q$	$\neg P \wedge Q$	$\neg P \wedge \neg Q$
T	T	T	F	F	F
T	F	F	T	F	F
F	T	F	F	T	F
F	F	F	F	F	T

For every truth value T in the truth table of the given formula, select the minterm which also has the value T for some combination of the truth values of P and Q.

SOLVED EXAMPLES

Example 1

The truth table for $P \Rightarrow Q$, $P \vee Q$ and $\neg(P \wedge Q)$ are given in the table below. Obtain the principal disjunctive normal forms of these formulae.

P	Q	$P \Rightarrow Q$	$P \vee Q$	$\neg(P \wedge Q)$
T	T	T	T	F
T	F	F	T	T
F	T	T	T	T
F	F	T	F	T

Solution

$$P \Rightarrow Q \Leftrightarrow (P \wedge Q) \vee (\neg P \wedge Q) \vee (\neg P \wedge \neg Q)$$
$$P \vee Q \Leftrightarrow (P \wedge Q) \vee (P \wedge \neg Q) \vee (\neg P \wedge Q)$$
$$\neg(P \wedge Q) \Leftrightarrow (P \wedge \neg Q) \vee (\neg P \wedge Q) \vee (\neg P \wedge \neg Q)$$ ■

Example 2

Obtain the principal disjunctive normal forms of

(a) $\neg P \vee Q$;
(b) $(P \wedge Q) \vee (\neg P \vee R) \vee (Q \wedge R)$

Solution

$$\neg P \vee Q \Leftrightarrow (\neg P \wedge (Q \vee \neg Q)) \vee (Q \wedge (P \vee \neg P)) \quad (A \wedge T \Leftrightarrow A)$$
$$\Leftrightarrow (\neg P \wedge Q) \vee (\neg P \wedge \neg Q) \vee (Q \wedge P) \vee (Q \wedge \neg P)$$
(distributive laws)
$$\Leftrightarrow (\neg P \wedge Q) \vee (\neg P \wedge \neg Q) \vee (P \wedge Q)$$
(commutative law and $P \vee P \Leftrightarrow P$)

(*b*) $(P \wedge Q) \vee (\neg P \wedge R) \vee (Q \wedge R)$
$$\Leftrightarrow (P \wedge Q \wedge (R \vee \neg R)) \vee (\neg P \wedge R) \wedge (Q \vee \neg Q) \vee (Q \wedge R) \wedge (P \vee \neg P))$$
$$\Leftrightarrow (P \wedge Q \wedge R) \vee (P \wedge Q \wedge \neg R) \vee (\neg P \wedge Q \wedge R) \vee (\neg P \wedge \neg Q \wedge R)$$ ■

Example 3

Show that the following are equivalent formulae

(a) $P \vee (P \wedge Q) \Leftrightarrow P$
(b) $P \vee (\neg P \wedge Q) \Leftrightarrow P \vee Q$

Solution

We write the principal disjunctive normal form of each formula and compare these normal forms.

$$(a)\ P \vee (P \wedge Q) \Leftrightarrow (P \wedge (Q \vee \neg Q)) \vee (P \wedge Q) \Leftrightarrow (P \wedge Q) \vee (P \wedge \neg Q)$$
$$P \Leftrightarrow P \wedge (Q \vee \neg Q) \Leftrightarrow (P \wedge Q) \vee (P \wedge \neg Q)$$
$$(b)\ P \vee (\neg P \wedge Q) \Leftrightarrow (P \wedge (Q \vee \neg Q)) \vee (\neg P \wedge Q)$$
$$\Leftrightarrow (P \wedge Q) \vee (P \wedge \neg Q) \vee (\neg P \wedge Q)$$
$$P \vee Q \Leftrightarrow (P \wedge (Q \vee \neg Q)) \vee (Q \wedge (P \vee \neg P)$$
$$\Leftrightarrow (P \wedge Q) \vee (P \wedge \neg Q) \vee (\neg P \wedge Q)$$ ■

Example 4

Obtain the principal disjunctive normal form of

$$P \Rightarrow ((P \Rightarrow Q) \wedge \neg(\neg Q \vee \neg P)$$

Solution

Using $P \Rightarrow Q \Leftrightarrow \neg P \Leftrightarrow Q$ and De Morgan's law, we obtain

$$P \Rightarrow (P \rightarrow Q) \wedge \neg(\neg Q \vee \neg P)$$
$$\Leftrightarrow \neg P \vee ((\neg P \vee Q) \wedge (Q \wedge P))$$
$$\Leftrightarrow \neg P \vee (\neg P \wedge (Q \wedge P)) \vee (Q \wedge (Q \wedge P))$$
$$\Leftrightarrow \neg P \vee (Q \wedge P)$$
$$\Leftrightarrow (\neg P \wedge (Q \vee \neg Q)) \vee (Q \wedge P)$$
$$\Leftrightarrow (\neg P \wedge P) \vee (\neg P \wedge \neg Q) \vee (P \wedge Q)$$ ■

3.11.4 Principal Conjuctive Normal Forms (PCNF)

If we are given a number of variables, max term of these variables is a formula consisting of disjunction in which each variable or its negation but not both appear only once. We observe that the maxterms are the duals of minterms. We note that each of minterms has the truth value *T* for exactly one combination of the truth values of the variables and this fact can be seen from truth table. Each of the maxterms has the truth value *F* for exactly one combination of the truth values of the variables. This fact can be derived from the duality principal or can be directly obtained from truth table.

For a given formula, an equivalent formula consisting of conjunctions of the maxterms only is called principal conjuctive normal form or the product-of-sums canonical form.

SOLVED EXAMPLES

Example 1

Obtain PCNF of $(\neg P \to R) \wedge (Q \Leftrightarrow P)$.

Solution

$$(\neg P \Rightarrow P) \wedge (Q \Leftrightarrow P)$$
$$\Leftrightarrow (P \vee R) \wedge (Q \to P) \wedge (P \to Q)$$
$$\Leftrightarrow (P \vee R) \wedge (\neg Q \vee P) \wedge (\neg P \vee Q)$$
$$\Leftrightarrow (P \vee R \vee (Q \wedge \neg Q)) \wedge (\neg Q \vee P \vee (R \wedge \neg R)) \wedge (\neg P \vee Q \vee (R \wedge \neg R))$$
$$\Leftrightarrow (P \vee Q \vee R) \wedge (P \vee \neg Q \vee R) \wedge (P \vee \neg Q \vee \neg R) \wedge (\neg P \vee Q \vee R) \wedge (\neg P \vee Q \vee \neg R)$$

Now the conjunctive normal form of $\neg S$ can easily be obtained by writing the conjunction of the remaining maxterms; thus, $\neg S$ has the principal conjunctive normal form

$$(P \vee Q \vee \neg R) \wedge (\neg P \vee \neg Q \vee R) \wedge (\neg P \vee \neg Q \vee \neg R)$$

By considering $\neg\neg S$, we obtain

$$\neg(P \vee Q \vee \neg R) \vee \neg(\neg P \vee \neg Q \vee R) \vee \neg(\neg P \vee \neg Q \vee \neg R)$$
$$\Leftrightarrow (\neg P \wedge \neg Q \wedge R) \vee (P \wedge Q \wedge \neg R) \vee (P \wedge Q \wedge R)$$

which is the principal disjunctive normal form of S. ■

Example 2

Show that the following are equivalent formulae

$$(i)\ P \vee (P \wedge Q) \Leftrightarrow P \quad (ii)\ P \vee (\neg P \wedge Q) \Leftrightarrow P \vee Q.$$

Solution

$$(i)\ P \Leftrightarrow P \wedge (Q \vee \neg Q) \Leftrightarrow (P \wedge Q) \vee (P \wedge \neg Q)$$
$$P \vee (P \wedge Q) \Leftrightarrow (P \wedge (Q \vee \neg Q)) \vee (P \wedge Q)$$
$$\Leftrightarrow (P \wedge Q) \vee (P \wedge \neg Q).$$

(*ii*) Try yourself (from above) ■

Example 3

Without constructing the truth tables, find the principal conjunctive normal forms of the following statements:

$$(a)\ (P \wedge Q) \vee (\neg p \wedge q \wedge r)$$

(b) $(p \vee q) \wedge (r \vee \neg p) \wedge (q \vee \neg r)$
(c) $(p \vee \neg(q \vee r)) \vee (((p \wedge q) \wedge \neg r) \wedge p)$
(d) $(p \to (q \wedge r)) \wedge (\neg p \to (\neg q \wedge \neg r)$

Solution

$(a)\ (p \wedge q) \vee (\neg p \wedge q \wedge r) \equiv ((p \wedge q) \vee \neg p) \wedge ((p \wedge q) \vee q) \wedge ((p \wedge q) \vee r)$

$\equiv (p \vee \neg p) \wedge (q \vee \neg p) \wedge (p \vee q) \wedge (q \vee q) \wedge (p \vee r) \wedge (q \vee r)$

$\equiv T \wedge (\neg p \vee q) \wedge (p \vee q) \wedge q \wedge (p \wedge r) \wedge (q \vee r)$

$\equiv ((\neg p \vee q) \vee (r \wedge \neg r)) \wedge ((p \vee q) \vee (r \wedge \neg r)) \wedge q \vee (p \wedge \neg p)$
$\wedge (p \vee r) \vee (q \wedge \neg q) \wedge (q \vee r) \vee (p \wedge \neg p) \qquad (\because \quad A \vee F = A)$

$\equiv (\neg p \vee q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (p \vee q \vee r) \wedge (p \vee q \vee \neg r)$
$\wedge (q \vee p) \wedge (q \vee \neg p) \wedge (p \vee r \vee q) \wedge (p \vee r \vee \neg q) \wedge (q \vee r \vee p)$
$\wedge (q \vee r \vee \neg p)$

$\equiv (\neg p \vee q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (p \vee q \vee r) \wedge (p \vee q \vee \neg r)$
$\wedge (p \vee \neg q \vee r) \wedge ((q \vee p) \vee (r \wedge \neg r)) \wedge ((q \vee \neg p) \vee (r \vee \neg r))$
(Omitting repetitions)

$\equiv (\neg p \vee q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (p \vee q \vee r) \wedge (p \vee q \vee \neg r)$
$\wedge (p \vee \neg q \vee r) \quad ...(1)$

(b) Let $S \equiv (p \vee q) \wedge (r \vee \neg p) \wedge (q \vee \neg r)$

Already S is in the CNF. Hence we can get the PCNF directly quickly.

$S \equiv ((p \vee q) \vee (r \wedge \neg r) \wedge ((\neg p \vee r) \vee (q \wedge \neg q)) \wedge ((q \vee \neg r) \vee (p \wedge \neg p))$

$\equiv (p \vee q \vee r) \wedge (p \vee q \vee \neg r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee \neg q \vee r)$
$\wedge (p \vee q \vee \neg r) \wedge (\neg p \vee q \vee \neg r)$

$\equiv (p \vee q \vee r) \wedge (p \vee q \vee \neg r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee \neg q \vee r)$
$\wedge (\neg p \vee q \vee \neg r)$

(c) Let $S \equiv (p \vee \neg(q \vee r)) \vee ((p \wedge q) \wedge \neg r) \wedge p$

$\equiv (p \vee (\neg q \vee \neg r)) \vee (p \wedge q \wedge \neg r \wedge p)$

$\equiv p \wedge (q \vee \neg q) \vee (\neg q \wedge \neg r) \vee (p \wedge q \wedge \neg r)$

$\equiv (p \wedge q) \vee (p \wedge \neg q) \vee (\neg q \wedge \neg r) \vee (p \wedge q \wedge \neg r)$

$\equiv ((p \wedge q) \wedge (r \vee \neg r)) \vee ((p \wedge \neg q) \wedge (r \vee \neg r)) \vee ((\neg q \wedge \neg r)$
$\wedge (p \vee \neg p)) \vee (p \wedge q \wedge \neg r)$

$\equiv (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge r) \vee (p \wedge \neg q \wedge \neg r)$
$\vee (p \wedge \neg q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge \neg r) \vee (p \wedge q \wedge \neg r)$

$\equiv (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge r)$

$\equiv (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge \neg q \wedge r) \vee (p \wedge \neg q \wedge \neg r)$
$\vee (\neg p \wedge \neg q \wedge r) \quad ...(1)$

In (1), we have got the PDNF of S.

Now $\neg S \equiv (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge r)$

$\therefore \qquad S \equiv \neg\neg S \equiv (p \vee \neg q \vee \neg r) \wedge (p \vee \neg q \vee r) \wedge (p \vee q \vee \neg r) \qquad ...(2)$

(2) is the required PCNF of S.

Let $S \equiv (p \to (q \wedge r)) \wedge (\neg p \to (\neg q \wedge \neg r))$

$\equiv (\neg p \vee (q \wedge r)) \wedge (p \vee (\neg q \wedge \neg r))$

$\equiv (\neg p \vee q) \wedge (\neg p \vee r)) \wedge (p \vee \neg q) \wedge (p \vee \neg r)$

$\equiv ((\neg p \vee q) \vee (r \wedge \neg r)) \wedge ((\neg p \vee r) \vee (q \wedge \neg q)) \wedge ((p \vee \neg q) \vee (r \wedge \neg r)) \wedge ((p \vee \neg r) \vee (q \wedge \neg q))$

$\equiv (\neg p \vee q \vee r) \wedge (\neg p \vee q \vee \neg r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee \neg q \vee r) \wedge (p \vee \neg q \vee r) \wedge (p \vee \neg q \vee \neg r) \wedge (p \vee q \vee \neg r) \wedge (p \vee \neg q + \neg r)$

$\equiv (p \vee \neg q \vee r) \wedge (p \vee \neg q \vee \neg r) \wedge (p \vee q \vee \neg r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee \neg q \vee r) \wedge (\neg p \vee q \vee \neg r)$ ■

3.12 Rules of Inference

This is also called Rules of P and T. In this article we shall describe the method by which on derives that a formula is a consequence of a set of given premises (hypothesis). The two rules inference is called the rule P and T as under.

Rule P: A promise may be introduced at any point of the derivation.

Rule T: A formula S may be introduced in a derivation if S is tautologically implied by any one or more of the preceding formulae in the derivation.

List of Implications

(*i*) **Implication table**

$I_1\ P \wedge Q \Rightarrow P$
$I_2\ P \wedge Q \Rightarrow Q$ } (simplification)

$I_3\ P \Rightarrow P \vee Q$
$I_4\ Q \Rightarrow P \vee Q$ } (addition)

$I_5\ \neg P \Rightarrow P \to Q$

$I_6\ Q \Rightarrow P \Rightarrow Q$

$I_7\ \neg(P \Rightarrow Q) \Rightarrow P$

$I_8\ \neg(P \Rightarrow Q) \Rightarrow \neg Q$

$I_9\ P, Q \Rightarrow P \wedge Q$

$I_{10}\ \neg P, P \vee Q \Rightarrow Q$ (disjunctive syllogism)

$I_{11}\ P, P \Rightarrow Q \Rightarrow Q$ (modus ponens)

$I_{12}\ \neg Q, P \Rightarrow Q \Rightarrow \neg P$ (modus tollens)

$I_{13}\ P \to Q, Q \to R \to P \Rightarrow R$ (hypothetical syllogism)

$I_{14}\ P \vee Q, P \Rightarrow R, Q \Rightarrow R \Rightarrow R$ (dilemma)

(*ii*) **Equivalence Table**

E_1	$\neg\neg P \Leftrightarrow P$	(Double negation)
E_2	$P \wedge Q \Leftrightarrow Q \wedge P$	(commutative laws)
E_3	$P \vee Q \Leftrightarrow Q \vee P$	
E_4	$(P \wedge Q) \wedge R \Leftrightarrow P \wedge (Q \wedge R)$	(associative laws)
E_5	$(P \vee Q) \vee R \Leftrightarrow P \vee (Q \vee R)$	
E_6	$P \wedge (Q \vee R) \Leftrightarrow (P \wedge Q) \vee (P \wedge R)$	(distributive laws)
E_7	$P \vee (Q \wedge R) \Leftrightarrow (P \vee Q) \wedge (P \vee R)$	
E_8	$\neg(P \wedge Q) \Leftrightarrow \neg P \vee \neg Q$	(De Morgan's laws)
E_9	$\neg(P \vee Q) \Leftrightarrow \neg P \wedge \neg Q$	
E_{10}	$P \vee P \Leftrightarrow P$	
E_{11}	$P \wedge P \Leftrightarrow P$	
E_{12}	$R \vee (P \wedge \neg P) \Leftrightarrow R$	
E_{13}	$R \wedge (P \vee \neg P) \Leftrightarrow R$	
E_{14}	$R \vee (P \vee \neg P) \Leftrightarrow T$	
E_{15}	$R \wedge (P \wedge \neg P) \Leftrightarrow F$	
E_{16}	$P \Rightarrow Q \Leftrightarrow \neg P \vee Q$	
E_{17}	$\neg(P \Rightarrow Q) \Leftrightarrow P \wedge \neg Q$	
E_{18}	$P \Rightarrow Q \Leftrightarrow \neg Q \Rightarrow \neg P$	
E_{19}	$P \Rightarrow (Q \Rightarrow R) \Leftrightarrow (P \wedge Q) \Rightarrow R$	
E_{20}	$\neg(P \Leftrightarrow Q) \Leftrightarrow P \Leftrightarrow \neg Q$	
E_{21}	$P \Leftrightarrow Q \Leftrightarrow (P \Rightarrow Q) \wedge (Q \Rightarrow P)$	
E_{22}	$(P \Leftrightarrow Q) \Leftrightarrow (P \wedge Q) \vee (\neg P \wedge \neg Q)$	■

3.13 Automatic Proving System (Theorems)

The concludive discussions/decisions are taken through automatic proving systems, which are as under:

(*i*) **Variables:** The capital letters of English alphabet *A*, *B*, *C*, ..., *P*, *Q*, *R*, ... will be used to denote the statement variables. They are also used to represent the statement formulas, but these notations will be clear from the context.

(*ii*) **Connectives:** The connectives $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\Leftrightarrow$ appear in the formulae with the order of precedence as given above.

(*iii*) **String of Formulae:** A string of formulae is defined as follows:

(*i*) any formula is a string of formulae,

(*ii*) if α and β are strings of formulae, then α, β and β, α are strings of formulae.

(*iii*) only those strings which are obtained from step (*i*) and (*ii*) are strings of formulae; with the exception of the empty string which is also a string of formulae.

The order in which the formulae appear in a string is not important; the strings *A*, *B*, *C*; *B*, *C*, *A*; *A*, *C*, *B* etc. are the same.

(*iv*) **Sequents:** If α and β are strings of formulae then $\alpha \xrightarrow{s} \beta$ is called a sequent in which α is the antecedent and β the consequent of the sequent.

A sequent $\alpha \xrightarrow{s} \beta$ is true if either at least one of the formulae of the antecedent is false or at least one of the formulae of the consequent is true. The former case is called the vacuous proof and the latter is called the Trivial proof.

Occasionally we have sequents which have empty strings of formulae as antecedent or as consequent. The empty antecedent is interpreted as the logical constant 'true' or *T* and the empty consequent is interpreted as the logical constant 'false' or *F*.

(*v*) **Axiom Scheme:** If α and β strings of formulae such that every formula in both α and β is a variable only, then the sequent $\alpha \xrightarrow{s} \beta$ is an axiom if α and β have atleast one variable in common. For example, $A, B, C \xrightarrow{s} P, B, R$ where *A*, *B*, *C*, *P*, *R* are variables is an axiom.

(*vi*) **Theorem:** The following sequents are theorems of our system.

(*i*) every axiom is a theorem,

(*ii*) if a sequent α is a theorem and a sequent β results from α through the use of one of the 10 rules of the system which are given below, then β is a theorem,

(*iii*) sequents obtained from (*i*) and (*ii*) are the only theorems.

(*vii*) Rules: The following rules are used to combine formulae within strings by introducing connectives.

There are two rules corresponding to each of the connectives, one for the introduction of the connective in the antecedent and the other for its introduction in the consequent. While describing these rules, α β γ, ... are used for strings of formulae while *X*, *Y* are used for formulae to which the connectives are applied.

(*a*) Antecedent Rules.

Rule $\neg$ $\Rightarrow$ If $\alpha, \beta \overset{s}{\Rightarrow} X, \gamma$, then $\alpha, \neg X, \beta \overset{s}{\Rightarrow} \gamma$.

Rule $\wedge$: If $X, Y, \alpha, \beta \overset{s}{\Rightarrow} \gamma$, then $\alpha, X \wedge Y, \beta \overset{s}{\Rightarrow} \gamma$.

Rule $\vee$: If $X, \alpha, \beta \overset{s}{\Rightarrow} \gamma$ and $Y, \alpha, \beta \overset{s}{\Rightarrow} \gamma$, then $\alpha, X \vee Y, \beta \overset{s}{\Rightarrow} \gamma$.

Rule $\Rightarrow \Rightarrow$: If $Y, \alpha, \beta \overset{s}{\Rightarrow} \gamma, \alpha, \beta \overset{s}{\Rightarrow} X, \gamma$, then $\alpha, X \Rightarrow Y, \beta \overset{s}{\Rightarrow} \gamma$.

Rule $\Leftrightarrow \Leftrightarrow$: If $X, \gamma, \alpha, \beta \overset{s}{\Rightarrow} \gamma, \alpha, \beta \overset{s}{\Rightarrow} X, Y, \gamma$, then $\alpha, X, \Leftrightarrow Y, \beta \overset{s}{\Rightarrow} \gamma$.

(*b*) Consequent Rules

Rule $\Rightarrow \neg$: If $X, \alpha, \overset{s}{\Rightarrow} \beta, \gamma$, then $\alpha \overset{s}{\Rightarrow} \beta, \neg X, \gamma$.

Rule $\Rightarrow$: If $\alpha \overset{s}{\Rightarrow} X, \beta, \gamma, \alpha \overset{s}{\Rightarrow} Y, \beta, \gamma$, then $\alpha \overset{s}{\Rightarrow} \beta, X \wedge Y . \gamma$.

Rule $\Rightarrow \vee$: If $\alpha \overset{s}{\Rightarrow} X, Y, \beta, \gamma$, then $\alpha \overset{s}{\Rightarrow} \beta, X \vee Y, \gamma$.

Rule $\Rightarrow \Rightarrow$: If $X, \alpha \overset{s}{\Rightarrow} Y, \beta, \gamma$, then $\alpha \overset{s}{\Rightarrow} \beta, X \Rightarrow Y, \gamma$.

Rule $\Rightarrow \Leftrightarrow$: If $X, \alpha \overset{s}{\Rightarrow} Y, \beta, \gamma$ and $Y, \alpha \overset{s}{\Rightarrow} X, \beta, \gamma$, then $\alpha \overset{s}{\Rightarrow} \beta, X \Leftrightarrow Y, \gamma$.

SOLVED EXAMPLES

Example 1

Demonstrate that R is a valid unference from the premises $P \Rightarrow Q$, $Q \Rightarrow R$, and P.

Solution

Step No.	Statement	Reason
[1]	(1) $P \Rightarrow Q$	Rule P
[2]	(2) P Rule P	
[1, 2]	(3) Q Rule T, (1), (2), and I_{11} (modus ponens)	
[4]	(4)	$Q \Rightarrow R$, Rule P
[1, 2, 4]	(5)	R Rule T, (3), (4), and I_{11}.

Illustration: The second column of numbers designates the formula as well as the line of derivation in which it occurs. The set of numbers in braces (the first column) for each line shows the premises on which the formula is the line depends. On the right, *P* or *T* represents the rule of unference followed by a comment showing from which formulae and tautology that particular formula has been obtained *e.g.,* if we follow this

notation, the third line shows that the formula in this line is numbered (3) and has been obtained from premises in (1) and (2). The comment on the right shows that the formula Q has been introduced using rule T and also indicates the details of the application of rule T. ■

Example 2

Show that $R \vee S$ follows logically from the premises $C \vee D$, $(C \vee D) \Rightarrow \neg H$, $\neg H \Rightarrow (A \wedge \neg B)$, and $(A \wedge \neg B) \Rightarrow (R \vee S)$.

Solution

Step No.	Statement	Reason
[1]	(1) $(C \vee D) \rightarrow \neg H$	P
[2]	(2) $\neg H \rightarrow (A \wedge \neg B)$	P
[1, 2]	(3) $(C \vee D) \rightarrow (A \wedge \neg B)$	T, (1), (2), and I_{13}
[4]	(4) $(A \wedge \neg B) \rightarrow (R \vee S)$	P
[1, 2, 4]	(5) $(C \vee D) \rightarrow (R \vee S)$	T, (3), (4), and I_{13}
[6]	(6) $C \vee D$	P
[1, 2, 4, 6]	(7) $R \vee S$	T, (5), (6), and I_{11}

■

Example 3

Show that $(p \rightarrow q) \wedge (r \rightarrow s)$, $(q \rightarrow t) \wedge (s \rightarrow u)$, $\neg(t \wedge u)$ and $(p \rightarrow r) \Rightarrow \neg p$.

Solution

Step No.	Statement	Reason
[1]	$(p \rightarrow q) \wedge (r \rightarrow s)$	P
[2]	$p \rightarrow q$	T, 1 and simplification
[3]	$r \rightarrow s$	T, 1 and simplification
[4]	$(q \rightarrow t) \wedge (s \rightarrow u)$	P
[5]	$q \rightarrow t$	T, 4 and simplification
[6]	$s \rightarrow u$	T, 4 and simplification
[7]	$p \rightarrow t$	T, 2, 5 and hypothetical syllogism
[8]	$r \rightarrow u$	T, 3, 6 and hypothetical syllogism
[9]	$p \rightarrow r$	P
[10]	$p \rightarrow u$	T, 8, 9 and hypothetical syllogism
[11]	$\neg t \rightarrow \neg p$	T and 7
[12]	$\neg u \rightarrow \neg p$	T and 10
[13]	$(\neg t \vee \neg u) \rightarrow \neg p$	T, 11, 12, and $(a \rightarrow b)$, $(c \rightarrow b)$, $\Rightarrow (a \vee c) \rightarrow b$

[14]	$\neg(t \vee u) \rightarrow \neg p$	*T*, 13 and De Morgan's law
[15]	$\neg(t \wedge u)$	*P*
[16]	$\neg p$	*T*, 14, 15 and modus ponens.

■

Example 4

Give a direct proof for the implication $p \rightarrow (q \rightarrow s), (\neg r \vee p), q \Rightarrow (r \rightarrow s)$

Solution

Step No.	Statement	Reason
[1]	$\neg r \vee p$	*P*
[2]	$r \rightarrow p$	*T*, 1 and equivalence of (1)
[3]	$p \rightarrow (q \rightarrow s)$	*P*
[4]	$r \rightarrow (q \rightarrow s)$	*T*, 2, 3 and hypothetical syllogism.
[5]	$\neg r \vee (\neg q \vee s)$	*T*, 4 and equivalence of (4)
[6]	q	*P*
[7]	$q \wedge (\neg r \vee \neg q \vee s)$	*T*, 5, 6 and conjunction
[8]	$q \wedge (\neg r \vee s)$	*T*, 7, 8 and negation and domination laws
[9]	$\neg r \vee s$	*T*, 8 and simplification
[10]	$r \rightarrow s$	*T*, 9 and equivalence of (9) ■

Example 5

Show that b can be derived from the premises $a \rightarrow b, c \rightarrow b, d \rightarrow (a \vee c)$, *d, by the indirect method.*

Solution

Let us include $\neg b$ as an additional premise and prove a contradiction.

Step No.	Statement	Reason
[1]	$a \rightarrow b$	*P*
[2]	$c \rightarrow b$	*P*
[3]	$(a \vee c) \rightarrow b$	*T*, 1, 2 and equivalence
[4]	$d \rightarrow (a \vee c)$	*P*
[5]	$d \rightarrow b$	*T*, 3, 4 and hypothetical syllogism
[6]	d	*P*
[7]	b	*T*, 5, 6 and modus ponens
[8]	$\neg b$	*P* (additional)
[9]	$b \wedge \neg b$	*T*, 7, 8 and conjunction
[10]	F	*T*, 9 and negation law. ■

Example 6

Show that $S \vee R$ is tautologically implied by $(P \vee Q) \wedge (P \Rightarrow R) \wedge (Q \Rightarrow S)$

Solution

Step No.	Statement	Reason
[1]	(1) $(P \vee Q)$	P
[1]	(2) $\neg P \Rightarrow Q$	T, (1), E_1, and E_{16}
[3]	(3) $Q \Rightarrow S$	P
[1, 3]	(4) $\neg P \Rightarrow S$	T, (2), (3), and I_{13}
[1, 3]	(5) $\neg S \Rightarrow P$	T, (4), E_{18}, and E_1
[6]	(6) $P \Rightarrow R$	P
[1, 3, 6]	(7) $\neg S \Rightarrow R$	T, (5), (6), and I_{13}
[1, 3, 6]	(8) $S \vee R$	T, (7), E_{16} and E_1 ■

Example 7

Show that $R \wedge (P \vee Q)$ is a valid conclusion from the premises $P \vee Q$, $Q \Rightarrow R$, $P \Rightarrow M$, and $\neg M$.

Solution

Step No.	Statement	Reason
[1]	(1) $P \Rightarrow M$	P
[2]	(2) $\neg M$	P
[1, 2]	(3) $\neg P$	T, (1), (2), and I_{12}
[4]	(4) $P \vee Q$	P
[1, 2, 4]	(5) Q	T, (3), (4) and I_{10}
[6]	(6) $Q \Rightarrow R$	P
[1, 2, 4, 6]	(7) R	T, (5), (6) and I_{11}
[1, 2, 4, 6]	(8) $R \wedge (P \vee Q)$	T, (4), (7), and I_9 ■

Example 8

Prove that the premises $p \rightarrow q$, $q \rightarrow r$, $s \rightarrow \neg r$ and $p \wedge s$ are inconsistent.

Solution

If we derive a contradiction by using the given premises, it means that they are inconsistent.

Step No.	Statement	Reason
[1]	$p \rightarrow q$	P
[2]	$q \rightarrow r$	P

[3]	$p \to r$	T, 1, 2 and hypothetical syllogism
[4]	$s \to \neg r$	P
[5]	$r \to \neg s$	T, 4 and contrapositive
[6]	$q \to \neg s$	T, 2, 5 and hypothetical syllogism
[7]	$\neg q \vee \neg s$	T, 6 and equivalence of (6)
[8]	$\neg(q \wedge s)$	T, 7 and De Morgan's law
[9]	$q \wedge s$	P
[10]	$(q \wedge s) \wedge \neg(q \wedge s)$	T, 8, 9 and conjunction
[11]	F	T, 10 and negation law

Hence the given premises are inconsistent ■

Example 9

Show that $R \Rightarrow S$ can be derived from the premises $P \Rightarrow (Q \Rightarrow S)$, $\neg R \vee P$, and Q.

Solution

Instead of deriving $R \to S$, we shall include R as an additional premise and show S first.

Step No.	Statement	Reason
[1]	(1) $\neg R \vee P$	P
[2]	(2) R	P (assumed premise)
[1, 2]	(3) P	T, (1), (2), and I_{10}
[4]	(4) $P \Rightarrow (Q \Rightarrow S)$	P
[1, 2, 4]	(5) $Q \Rightarrow S$	T, (3), (4) and I_{11}
[6]	(6) Q	P
[1, 2, 4, 6]	(7) S	T, (5), (6), and I_{11}
[1, 4, 6]	(8) $R \to S$	CP ■

Example 10

Show that the following premises are inconsistent.

1. If Jasika misses many classes through illness, then he fails high school.

2. If Jasika fails high school, then he is uneducated.

3. If Jasika reads a lot of books, then he is not uneducated.

4. *Jasika misses many classes through illness and reads a lot of books.*

Solution

E : Jasika misses many classes.
S : Jasika fails high school.
A : Jasika reads a lot of books.
H : Jasika is uneducated.

The premises are $E \Rightarrow S$, $S \Rightarrow H$, $A \Rightarrow \neg H$, and $E \wedge A$.

Steps No.	Statement	Reason
[1]	(1) $E \Rightarrow S$	P
[2]	(2) $S \Rightarrow H$	P
[1, 2]	(3) $E \Rightarrow H$	T, (1), (2) and I_{13}
[4]	(4) $A \Rightarrow \neg H$	P
[4]	(5) $H \Rightarrow \neg A$	T, (4), E_{18}
[1, 2, 4]	(6) $E \Rightarrow \neg A$	T, (3), (5), I_{13}
[1, 2, 4]	(7) $\neg E \vee \neg A$	T, (6), E_{16}
[1, 2, 4]	(8) $\neg (E \wedge A)$	T, (7), E_8
[9]	(9) $E \wedge A$	P
[1, 2, 4, 9]	(10) $(E \wedge A) \wedge \neg (E \wedge A)$	T, (8), (9), I_9. ■

Example 11

Using indirect method of proof, derive $p \rightarrow \neg s$ *from the premises* $p \rightarrow (q \vee r)$, $q \rightarrow \neg p$, $s \rightarrow \neg r$; p.

Solution

Let us include $\neg (p \rightarrow \neg s)$ as an additional premise and prove a contradiction.
Now $\neg (p \rightarrow \neg s) = \neg (\neg p \vee \neg s) = p \wedge s$

Hence the additional premise to be introduced may be taken as $p \wedge s$.

Step No.	Statement	Reason
[1]	$q \rightarrow (q \vee r)$	P
[2]	p	P
[3]	$q \vee r$	T, 1, 2 and modus ponens
[4]	$p \wedge s$	P(additional)
[5]	s	T, 4 and simplification
[6]	$s \rightarrow \neg r$	P
[7]	$\neg r$	T, 5, 6 and modus ponens
[8]	q	T, 3, 7 and disjunctive syllogism
[9]	$q \rightarrow \neg p$	P
[10]	$\neg p$	T, 8, 9 and modus ponens
[11]	$p \wedge \neg p$	T, 2, 10 and conjunction
[12]	F	T, 11 and negation law

Example 12

Prove that the premises $a \to (b \to c)$, $d \to (b \wedge \neg c)$ and $(a \wedge d)$ are inconsistent.

Solution

Step No.	Statement	Reason
[1]	$a \wedge d$	P
[2]	a	T, 1 and simplification
[3]	d	T, 1 and simplification
[4]	$a \to (b \to c)$	P
[5]	$b \to c$	T, 2, 4 and modus ponens
[6]	$\neg b \vee c$	T, 5 and equivalence of (5)
[7]	$d \to (b \wedge \neg c)$	P
[8]	$\neg(b \wedge \neg c) \to \neg d$	T, 7 and contrapositive
[9]	$\neg b \vee c \to \neg d$	T, 8 and equivalence
[10]	$\neg d$	T, 6, 9 and modus ponens
[11]	$d \wedge \neg d$	T, 3, 10 and conjunction
[12]	F	T, 11 and negation law

Hence the given premises are inconsistent. ■

Example 13

Show that the following set of premises is inconsistent. If Seema gets her degree, she will go for a job. If she goes for a job, she will get married soon to Santosh. If she goes for higher study, she will not get married. Seema gets her degree and goes for higher study.

Solution

Let the statement be symbolic as follows:

p: Seema gets her degree.
q: she will go for a job.
r: she will get married soon
s: she goes for higher study.
Then we have to prove that
$p \to q, q \to r, s \to \neg r, p \wedge s$ are inconsistant.

Step No.	Statement	Reason
[1]	$p \to q$	P
[2]	$q \to r$	P

[3]	$p \to r$	T, 1, 2 and hypothetical syllogism
[4]	$p \to s$	P
[5]	p	T, 4 and simplification
[6]	s	T, 4 and simplification
[7]	$s \to \neg r$	P
[8]	$\neg r$	T, 6, 7 and modus ponens
[9]	r	T, 3, 5 and modus ponens
[10]	$r \wedge \neg r$	T, 8, 9 and conjunction
[11]	F	T, 10 and negation law

Hence the set of given premises is inconsistent. ■

Example 14

Construct an argument to show that the following promises imply the conclusion "it rained in JJT University."

"If it does not rain or if there is no traffic distraction, then the sports day will be held in JJT University and the cultural programme will go on", "If the sports day is held, the trophy will be awarded" and "the trophy was not awarded".

Solution

Let the symbolic statements will be:

p: It rains.
q: there is traffic dislocalion
r: sports day will be held.
s: cultural programme will go on
t: He trophy will be awarded.
Then it should be proved that
$\neg P \vee \neg q \to r \wedge s, r \to t, \neg t \Rightarrow p$

Step No.	**Statement**	**Reason**
[1]	$\neg p \vee \neg q \to r \wedge s$	P
[2]	$(\neg p \to (r \wedge s))$ $\wedge (\neg q \to (r \wedge s))$	T, 1 and the equivalence $(a \vee b) \to c \equiv (a \to c)$ $\wedge (b \to c)$
[3]	$\neg (r \wedge s) \to p$	T, 2 and contrapositive of (2)
[4]	$r \to t$	P
[5]	$\neg t \to \neg r$	T, 4 and contrapositive of (4)
[6]	$\neg t$	P
[7]	$\neg r$	T, 5, 6 and modus ponens

[8]	$\neg r \vee \neg s$	T, 7 and addition
[9]	$\neg(r \wedge s)$	T, 8 and De Morgan's law
	[10]	p T, 3, 9 and modus ponens ■

Example 15

Show that $P \vee Q$ following from P.

Solution

We need to show that

(1) $\Rightarrow P \overset{s}{\Rightarrow} (P \vee Q)$

(1) if (2) $P \overset{s}{\Rightarrow} P \vee Q \quad (\Rightarrow \rightarrow)$

(2) if (3) $P \overset{s}{\Rightarrow} P, Q \quad (\Rightarrow \vee)$

We first eliminate the connective $\rightarrow$ in (1). Using the rule $\Rightarrow \rightarrow$ we have "if $P \overset{s}{\Rightarrow} P \vee Q$ then $\overset{s}{\Rightarrow} P \rightarrow (P \vee Q)$." Here we have named $P \overset{s}{\Rightarrow} P \vee Q$ by (2). Each line of derivation thus introduces the name as well as gives a rule. Note also that "(1) if (2)" means "if (2) then (1)". The chain of arguments is then given by (1) holds (2), and (2) holds if (3). Finally (3) is a theorem, because it is an axiom. The actual derivation is simply a reversal of these step in which (3) in an axiom that leads to $\overset{s}{\Rightarrow} P \rightarrow (P \vee Q)$ as shown.

(*a*) $P \overset{s}{\Rightarrow} P, Q$	Axiom
(*b*) $P \overset{s}{\Rightarrow} P \vee Q$	Rule ($\Rightarrow \vee$), (*a*)
(*c*) $\overset{s}{\Rightarrow} P \rightarrow (P \vee Q)$	Rule ($\Rightarrow \rightarrow$), (*b*), ■

Example 16

Show that $\overset{s}{\Rightarrow} (\neg Q \wedge (P \Rightarrow Q)) \Rightarrow \neg P$.

Solution

(1) $\overset{s}{\Rightarrow} (\neg Q \wedge (P \Rightarrow Q)) \Rightarrow \neg P$

(1) if (2) $\neg Q \wedge (P \Rightarrow Q) \overset{s}{\Rightarrow} \neg P \quad (\Rightarrow \rightarrow)$

(2) if (3) $\neg Q, P \Rightarrow Q \overset{s}{\Rightarrow} \neg P \quad (\wedge \Rightarrow)$

(3) if (4) $P \to Q \overset{s}{\Rightarrow} \neg P, Q$ $(\neg \Rightarrow)$

(4) if (5) $Q \overset{s}{\Rightarrow} \neg P, Q$ and (6) $\overset{s}{\Rightarrow} P, \neg P, Q$ $(\to \Rightarrow)$

(5) if (7) $P, Q \overset{s}{\Rightarrow} Q$ $(\Rightarrow \neg)$

(6) if (8) $P \overset{s}{\Rightarrow} P, Q$ $(\Rightarrow \neg)$

Now (7) and (8) are axioms, hence the theorem (1) follows. We omit the derivation, which is easily obtained by starting with the axioms (7) and (8) and retracing the steps. ■

Example 17

Does P follow from $P \vee Q$?

Solution

We investigate whether $\overset{s}{\Rightarrow} (P \vee Q) \to P$ is a theorem.

Assume (1) $\overset{s}{\Rightarrow} (P \vee Q) \Rightarrow P$.

(1) if (2) $P \vee Q \overset{s}{\Rightarrow} P$ $(\Rightarrow \to)$

(2) if (3) $P \overset{s}{\Rightarrow} P$ and (4) $Q \overset{s}{\Rightarrow} P$ $(\vee \Rightarrow)$

Note that (3) is an axiom, but (4) is not. Hence P does not follow from $P \vee Q$. ■

Example 18

If L(x, y) symbolires the statement "x loves y", where the universe of discourse for both x and y consists of all people in the world, translate the following English sentences into logical impression:

(a) Every body loves z.
(b) Every body loves somebody:
(c) There is somebody whom everybody loves.
(d) Nobody loves everybody.
(e) There is somebody whom no one loves.

Solution

(*a*) $L(x, z)$ for all x. Hence $\forall x\, L(x, z)$

(*b*) $L(x, y)$ is true for all x and some y. Hence $\forall \exists y\, L(x, y)$

(*c*) Eventhough, (*c*) is the same as (*b*), the stress is on the existence of somebody (*y*) whom all *x* love.

Hence $\exists\, y\, \forall\, x\, L(x, y)$

(*d*) Nobody loves every body

i.e., There is not one who loves everybody

Hence $\neg\ \exists x\ \forall y\ L(x, y)$

$\equiv \forall x\ \neg \forall y\ L(x, y)$

$\equiv \forall x\ \exists y\ \neg L(x, y)$

(*e*) The sentence means that there is somebody whom every one does not love.

Hence $\neg\ \forall x\ \exists y\ L(x, y)$

$\equiv \exists x\ \neg \exists y\ L(x, y)$

$\equiv \exists x\ \forall y\ \neg L(x, y)$ ■

Example 19

Show that $\neg (P \wedge Q)$ *follows from* $\neg P \wedge \neg Q$.

Solution

We introduce $\neg\neg (P \wedge Q)$ as an additional premises and show that this additional premise leads to a contradiction.

Steps No.	Statement	Reason
[1]	(1) $\neg\neg (P \wedge Q)$	P (assumed)
[1]	(2) $P \wedge Q$	T, (1) and E_1
[1]	(3) P	T, (2), and I_1
[4]	(4) $\neg P \wedge \neg Q$	P
[4]	(5) $\neg P$	T, (4), I_1
[1, 4]	(6) $P \vee \neg P$	T, (3), (5), I_9

■

3.14 The Predicate Calculus

In the study of symbolic logic we also study statement calculous. In statement calculus we consider statements as basic units, we do not analyses any atomic statement. Only analysis of compound formulae are done in statement calculus. The statements are called proposition and statement calculus is also called propositions calculus.

We do not make any analysis of atomic statements in the study of statement calculous it was not possible to compare any two atomic state-

ments and it was not possible to know whether any two atomic statements have some features in common. The logic based upon the analysis of predicates in any statement is called *predicative logic.*

We consider the two statements

Ram is a bachelor.

Shyam is a babchelor.

In each statement 'is a bachelor' is a **predicate**. These two statements are about two different persons, but both have same predicate 'is a bachelor'.

We introduce a symbol to denote the predicate 'is a bachelor' and a method to connect it with the names of persons. Then we will have symbols to denote the statements of persons being bachelor.

Consider another well known and commonly used example.

All mean are martial.

Ram is a man.

Therefore, Ram is mortal.

Any human being will agree that a above conclusion is true so far s his intuition is concerned. But it as does not follow from the (Inference theory of) statement calculus In this case 'are mortal'or 'is mortal' is the predicate. When we separate out 'are mortal' form 'all men' and use different symbols for them, then it will be possible to draw conclusion about any particular man.

While constructing symbols, we shall use capital letters for the predicatives and small letters for the objects. For example, let B denote the predicate 'is a bachelor', let r, s respectively stand for Ram and Shyam.

Then

$B(r)$: Ram is a bachelor.

$B(s)$: Ram is a bachelor.

In general is a statement is given by 'P is Q' where Q is the predicate and P is the subject (object), then symbolically, it will be denoted by $Q(P)$.

Sometimes there are more than on subject associated with the same predicate. A predicate requiring m(m is ap positive integer) objects is called an m-place predict. Similarly 1-place predicate and 2-place predicate etc. are now self-explanatory.

Consider the following statements:

Ram is younger than Shyam.

Delhi is the north of Hyderabad.

These are examples of 2-place predicates. In the first statement 'is younger than, is the predicate, let us denote it by Y. As mentioned earlier,

then r, s respectively denote Ram and Shyam. Then the first statement will be represented by $Y(r, s)$. Similarly in the second statement, let N denote the predicate 'to the north of', let d and h respectively denote Delhi and Hyderabad. Then the second statement will be denoted by $N(d, h)$.

3.14.1 Statement Functions, Variables and Quantifiers

To clarify the formal definition of statement function, we take an example. Let M denotes the predicate is mortal'. Let r, d and t denote Ram, Delhi and table respectively. We can say that $M(r)$, $M(d)$ and $M(t)$ are statements.

Let $M(x)$ denote that 'X is mortal'

Note that $M(x)$ is not attachment. The statements $M(r)$, $M(m)$, $M(t)$ follow from $M(x)$ by replacing the variable x respectively by and r, d and t. In general $M(x)$ will given us a statement when the variable x is replaced by an object . *W*e now give a formal definition.

"A simple statement function in one variable is an expression consisting of a predicate symbol and a single variable." Such a statement function becomes a statement when the variable is replaced by a definite object. **"The statement resulting from a replacement is called a substitution instance of the statement function and is a formula of statement calculus." A compound statement function is obtained from combining one or more simple statement function and the logical connectives.**

For example, let $H(x)$ denotes 'x is a human being' and $M(x)$ denote 'x is mortal'. Then the following are some examples of compound statement function:

$$H(x) \wedge M(x),\ H(x) \quad \Rightarrow \quad M(x),\ \neg M(x),\ H(x) \vee \neg M(x)$$

3.14.2 Free and Bound Variables

With the help of well-formed formulae (wff), we have to discuss two formulae containing a part of form $\forall\ x\ P(x)$ or $\exists x\ P(x)$. This language consists of the connectives $\forall$, besides the other connectives.

A letter x is said to be *bound* on a string S if $\forall\ x$ or $\exists$ appear in S; otherwise x **bound free** or is free in S if x does not appear in S.

A string containing a part of the form $\forall$, $\exists$ besides the other connectives is called a formula off it satisfies the following properties:

(*i*) If P is a formula and x is a letter free in P, then $\forall x\ P$ is a formula.

(*ii*) If P is a formula and x is a letter free in P, then $\exists x\ P$ is a formula.

In fact $\forall xP\ (x) \vee Q\ (x)$ can be expressed in English as 'every x has the property P, and has the property Q'. A change in variables in the bound

occurrence is not a substitution instance. Sometimes it becomes necessary to change the variables to avoid confusion. It is better to write $\forall y\ P(y) \vee Q(x)$ instead of $\forall x\ P(x) \wedge Q(x)$, so as to separate the free and bound variables. Occasionally, one may come across a formula of the type $\forall\ x\ P(x)$ in which the occurrence of y is free and the shope of $\forall\ x$ does not contain x; in such a case, we have a vacuous use of $\forall\ x$.

It should be noted that in a statement every occurrence of a variable must be bound, and no variable should have a free occurrence. In the case where a free variable occurs in a formula, then we have a statement function.

As example consider the following formulas:

$\forall\ x\ P(x, y)$ ...(1)

$\forall\ x\ P(x) \Rightarrow \quad Q(x)$ ...(2)

$\forall\ xP(x) \Rightarrow \quad (\exists y)\ R(x, y)$ ...(3)

$\forall x\ P(x) \Rightarrow \quad R(x) \vee \forall\ x(P(x) \quad \Rightarrow \quad Q(x)).$ ...(4)

$(\exists x)\ (P(x) \wedge \quad Q(x)$ ...(5)

$(\exists x)\ P(x)\ \wedge \quad Q(x)$ ...(6)

In (1) both occurrences of x are bound, while the variable y is free. In (2), all occurrences of x are bound. In (3) all occurrences of both x and y are bound, in (4) all occurrences of x are bound. In (6), the last occurrence. of x in $Q(x)$ is free.

In the bound occurrence of a variable, the letter used to represent the variable is not important; any other letter can be used as a variable without affective the formula, provided that the new letter is not used elsewhere in the formula. Thus.

$$\forall\ x\ P(x, y) \text{ and } \forall\ z\ P(z, y)$$

are the same. It should also be noted that the bound occurrence of a variable cannot be substituted by a constant only a free occurrence of a variable can be. For example.

$$\forall\ x\ P(x) \wedge Q\ (a)$$

is a substitution instance of

$$\forall\ x\ P(x, y) \wedge Q(z, y).$$

3.14.3 Special Valid Formulae using Quantifiers

Let *A*(*x*) be a predicate formula where x is a object variable of interest.

Then we may white

$$\forall\ x\ A(x) \quad \Rightarrow \quad A(y)$$

where y is substituted for x in $A(x)$ to obtain $A(y)$. We assume that $\forall\ x\ A(x)$ is true. Then $A(y)$ is true. If $\forall\ x\ A(x)$ is false, then nothing need to be proved. This impaction can be written in a more convenient from as

$$\forall x A(x) \quad \Rightarrow \quad A(x).$$

This is called the rule of universal specification (US). It states that from a universally valid premise $\forall\, x\, A(x)$, we can conclude a specific case $A(x)$, so that one can drop the quantifier in the derivation.

We shall now discuss another implication. For this, let x by any object variable, B be any formula not containing any free occurrence of x and $A(x)$ be any formula.

Then

$$(B \Rightarrow A(x)) \quad \Rightarrow \quad (B \Rightarrow \forall\, x\, A(x)) \qquad ...(1)$$

This state that for an arbitrary object variable x if $A(x)$ follows from B, then we can conclude that $\forall\, x\, A(x)$ follows from B.

Note that if P, Q and R are statement formula then

$$(P + \neg P) \quad \Rightarrow \quad Q) \quad \Rightarrow \quad (P \vee \neg P) \quad \Rightarrow R) \quad \Rightarrow (Q \Rightarrow R) \qquad ...(2)$$

As a special case of (1) let $B = P \vee \neg P$, where P is any statement variable. Since $P \vee \neg P$ is a tautology, from

$$((P \vee \neg P) \Rightarrow \qquad A(x) \Rightarrow \quad ((P \vee \neg P) \quad \Rightarrow \quad \forall\, xA\,(x))$$

by using (2) we get

$$A(x) \Rightarrow \forall\, x\, A(x).$$

This permits us to conclude $\forall\, x\, A(x)$ from $A(x)$, so that we can add the universal quantifier in the conclusion during the course of derivation. This rule is called the rule of **universal generalization (UG)**

Similar to US and UG, there are two more rules which permit us to remove or include the existential quantifier during the course of derivation. They are

$$(\forall x)\, A(x) \quad \Rightarrow \quad A(y)$$

$$A(x) \qquad \Rightarrow \quad (\forall x)\, A(x)$$

They are respectively called as **existential specification** (ES) and existential generalization (EG).

3.14.4 Theory of Inference for the Predicate Calculus

If the conclusion of statement as given in the form of conditional, we also use the rule *CP.* We also use US, ES, UG and EG, which are described as under.

While describing the rules $A(x)$ will be used to denote a formula with a free occurrence of x $A(y)$ will denote a formula obtained by the substitution of y for x in $A(x)$, for such a substitution $A(x)$ must be free from y.

Rules US: From $\forall x A(x)$ one concludes $A(y)$.

Rules ES: From $(\exists x)\, A(x)$ one can conclude $A(y)$ provide that y is not free in any given premise and also not free in any prior step of the derivation. These requirements can easily by met by choosing a new variable each time ES is used. (The conditions of ES are more restrictive than ordinarily required, but they do not affect the possibility of deriving any conclusion).

Rule EG: From $A(x)$ one can conclude $(\forall\, y)\, A(y)$.

Rule UG: From $A(x)$ one can conclude $(\forall\, y)\, A(y)$ provided that x is not free in any of the given premises and provided that if x is free in a prior step which resulted from the use of ES, then no variable introduced by that use of ES appear free in $A(x)$.

3.14.5 Formulae Involving More than One Quantifier

When the qunantifiers occur in the form of combinations, these are possible even in the care of one place predicates and the become particularly important in the case of n-places predicates ($n \geq 2$). For example, if $P(x, y)$ is a 2-place predicate formula, then the following possibilities occur:

$(x)\ (y)\ P(x, y)$ $\quad$ $(x)\ (\exists y)\ P(x, y)$ $\quad$ $(\exists x)\ (y)\ P(x, y)$

$(\exists x)\ (\exists y)\ P(x, y)$ $\quad$ $(y)\ (x)\ P(x, y)$ $\quad$ $(\exists y)\ (x)\ P(x, y)$

$(y)\ (\exists x)\ P(x, y)$ $\quad$ $(\exists y)\ (\exists x)\ P(x, y)$

Since $(x)\ (y)\ P(x, y)$ stands for $(x)\ ((y)\ P(x, y))$ and $(\exists x)\ (y)\ P(x, y)$ for $(\exists x)\ ((y)\ P(x, y)$. The brackets are not used because without them there is no possibility of misunderstanding the meaning. From the meaning of the quantifiers, the following formulae can be obtained.

$$(x)\ (y)\ P(x, y) \Leftrightarrow (y)\ (x)\ P(x, y) \quad ...(1)$$

$$(x)\ (y)\ P(x, y) \Rightarrow (\exists y)\ (x)\ P(x, y) \quad ...(2)$$

$$(y)\ (x)\ P(x, y) \Rightarrow (\exists x)\ (y)\ P(x, y) \quad ...(3)$$

$$(\exists y)\ (x)\ P(x, y) \Rightarrow (x)\ (\exists y)\ P(x, y) \quad ...(4)$$

$$(\exists x)\ (y)\ P(x,y) \Rightarrow (y)\ (\exists x)\ P(x, y) \quad ...(5)$$

$$(x)\ (\exists y)\ P(x, y) \Rightarrow (\exists y)\ (\exists x)\ P(x, y) \quad ...(6)$$

$$(y)\ (\exists x)\ P(x, y) \Rightarrow (\exists x)\ (\exists y)\ P(x, y) \quad ...(7)$$

$$(\exists x)\ (\exists y)\ P(x, y) \Leftrightarrow (\exists y)\ (\exists x)\ P(x, y) \quad ...(8)$$

Figure below shows implications (2) to (7) and equivalences (1) and (8). One can also are prove these implication and equivalences using the method of derivation given in the previous section.

The negation of any of the above formulae can be obtained by repeated applications of the equivalences E_{25} and E_{26} For example,

$$\neg(\exists y)\ (x)\ P(x, y) \Leftrightarrow (y)\ (\neg(x)\ P(x, y)) \Leftrightarrow (y)\ (\exists x)\ \neg P(x, y)$$

The negations of other formulas of this type are obtained in a similar manner.

The inference rule and the method of derivation as given earlier also apply to *n*-place predicate formulae. Obviously, some special care would now be needed in the use of the rules UG, EG, US, and ES. In the case of US and ES, the specific variable should be chosen in such a way that it is different from the bound variable used elsewhere. To illustrate this discussion, we consider the formula $(x)\ (\exists y)\ P(x, y)$. Using US, we can write any of the formulae.

$(\exists y)\ P(x, y)$, $(\exists y)\ P(z, y)$

but we should not write $(\exists y)\ P(y, y)$ because the variable y is used as a bound variable: that is, $(\exists y)\ P(x, y)$ is not free for y. Similarly, in using EG, on should be careful. For example, from $(x)\ P(x, y)$ we can generalize.

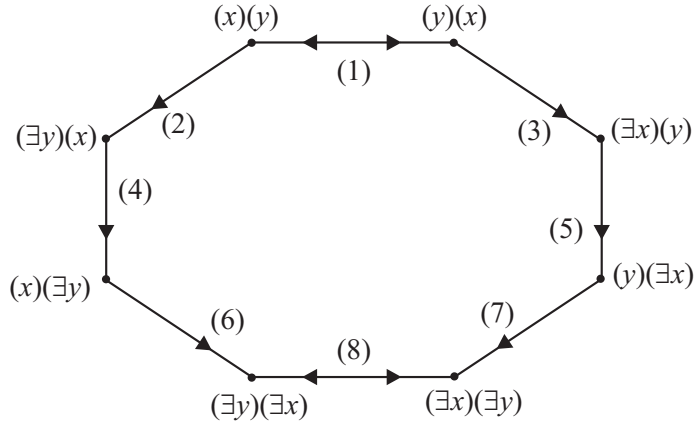


Fig. 3.5: Graphical representation of relationships among formulas involving two quantifiers.

$(\exists y)\ (x)\ P(x, y)$ or $(\exists z)\ (x)\ P(x, z)$

but not $(\exists x)\ (x)\ P(x, x)$. Similar care is required in the use of UG and *ES.*

SOLVED EXAMPLES

Example 1

If *P(x): x is a person.*

F(x, y): x is the father of y

M(x, y): x is the mother of y.

write the predicate "x is the father of the mother of y."

Solution

To symbolize the predicate, we name a person z as the mother of y it is obvious that we want to say that z is the father of z and z the mother of y. We assume that such a person z exists. We can symbolize the predicate as

$$(\exists z)\,(P(z) \wedge F(x, z) \wedge M(z, y).$$ ■

Example 2

Symbolize the expression "All the world loves a lady".

Solution

First we note that the quotation really means that everybody loves a lady.

Let $P(x)$: x is a person.
$L(x)$: y is a lady.
$R(x, y)$: x loves y.

The required expression is

$$(x)\,(P(x) \rightarrow (y)\,(P(y) \wedge L(y) \rightarrow R(x, y)))$$ ■

Example 3

Consider the statement:

All men are mortal
Santosh is man.
Therefore Santosh is mortal.

Solution

Let us use the notations as:

$H(x)$: x is a man.
$M(x)$: x is mortal
s: Santosh.

with the above symbols, the problem becomes.

$$\forall\, x(H(x) \rightarrow M(x) \wedge H(s) \quad \Rightarrow \quad M(s)$$

The derivation of the proof is,

Step No.	Statement	Reason
1	$\exists\, x(H(x) \rightarrow M(x)$	P
2	$H(s) \rightarrow M(s)$	*US,* 2
3	$H(s)$	P
4	$M(s)$	T, 2, 3, Modus ponens

Example 4

Consider the statement "Given any positive integer, there is a greater positive integer" Symbolize this statement with and without using the set of positive integers as the universe of discourse.

Solution

Consider the variables x and y be restricted to the stet of positive integers. Then the above statement can be paraphrased as follows: for all x, there exists a y such that y is greater than x.

If $G(x, y)$ is "x is greater than y", then the given statement is:

$$(x)\ (\exists y)\ G(y, x).$$

If we do not impose the restriction on the universe of discourse and if we write $P(x)$ for "x is a positive integer," then

$$(x)\ (P(x) \rightarrow (\exists y)\ (P(y) \wedge G(y, x))) \qquad \blacksquare$$

Example 5

Show that

$$(x)\ (P(x) \rightarrow Q(x) \wedge (x)\ (Q(x) \rightarrow R(x) \Rightarrow (x)\ (P(x) \rightarrow R(x))$$

Solution

Stop No.	Statement	Reason
[1]	(1) $(x)\ (P(x) \rightarrow Q(x)$	P
[1]	(2) $P(y) \rightarrow Q(y)$	US, (1)
[3]	(3) $(x)\ (Q(x) \rightarrow R(x)$	P
[3]	(4) $Q(y) \rightarrow R(y)$	US, (3)
[1, 3]	(5) $P(y) \rightarrow R(y)$	T, (2); (4), I_{13}
[1, 3]	(6) $(x)\ (P(x) \rightarrow R(x)$	UG, (5) $\blacksquare$

Example 6

Show that from

(a) $(\exists x)\ (F(x) \wedge S(x) \rightarrow (y)\ (M(y) \rightarrow W(y))$

(b) $(\exists y)\ (M(y) \wedge \neg W(y)$

the conclusion $(x)\ (F(x) \rightarrow \neg S(x))$ *follows.*

Solution

Stop No.	Statement	Reason
[1]	(1) $(\exists y)\ (M(y) \wedge W(y))$	P
[1]	(2) $M(z) \wedge \neg W(z)$	ES, (1)

[1]	(3) $\neg(M(z) \rightarrow W(z)$	T, (2), E_{17}
[1]	(4) $(\forall y) \wedge (y) \rightarrow W(y))$	EG, (3)
[1]	(5) $\neg(y)\,(M(y) \rightarrow W(y))$	
[6]	(6) $(\forall x)\,(F(x) \wedge S(x)$	$E_{26}(3)$
	$\rightarrow (y)(M(y) \rightarrow W(y))$	P
[1, 6]	(7) $(\exists x)\,(F(x) \wedge S(x))$	T, (5), (6), I_{12}
[1, 6]	$(8)(x) \wedge (F(x) \wedge S(x))$	T, (7), E_{25}
[1, 6]	(9) $\neg(F(x) \wedge S(x))$	US, (8)
[1, 6]	$(10)F(x) \rightarrow \neg S(x)$	T, (9), E_9, $E_{16}E_{17}$
[1, 6]	(11) $(x)\,(F(x) \rightarrow \neg\, S(x))$	UG, (10) ■

Example 7

Prove that

$$(\exists\, x)\,(P(x)\ \wedge Q(x)) \quad \Rightarrow \quad (\exists\, x)\,P(x) \wedge (\exists x)\,Q(x).$$

Solution

Stop No.	Statement	Reason
[1]	(1) $(\exists x)\,(P(x) \wedge Q(x)$	P
[1]	(2) $P(y) \wedge Q(y)$	ES, (1), y fixed
[1]	(3) $P(y)$	T, (2), I_1
[1]	(4) $Q(y)$	T, (2), I_2
[1]	(5) $(\exists x)\,P(x)$	EG, (3)
[1]	(6) $(\exists x)\,Q(x)$	EG, (4)
[1]	(7) $(\exists x)\,P(x) \wedge (\exists x)\,Q(x)$	T, (4), (5), I_9

It is instructive to try to prove the converse which does not hold. The derivation is

(1)	$(\exists x)\,P(x) \wedge (\exists x)\,Q(x)$	P
(2)	$(\exists x)\,P(x)$	T, (1), I_1
(3)	$(\exists x)\,Q(x)$	T, (1), I_2
(4)	$P(y)$	ES, (2)
(5)	$Q(z)$	ES, (3)

Note that in step 4, y is fixed, and it is no longer possible to use that variable again in step 5. ■

Example 8

Express the negations of the following statements using quantifiers and in English:

(a) If the teacher is absent, then some students do not keep quiet.
(b) All the students keep quiet and the teacher is present.

(c) Some of the students do not keep quiet or the teacher is absent.
(d) No one has done every problem in the exercise.

Solution

(*a*) Let T represent the presence of the teacher and $Q(x)$ represent "x keeps quiet".

Then the given statement is:

$$\neg T \to \exists x\, Q(x) \equiv \neg T \to \neg \forall x\, Q(x)$$
$$\equiv T \vee \neg \forall x\, Q(x)$$

$\therefore$ Negation of the given statement is

$$\neg (T \vee \neg \forall x\, Q(x)) \equiv \neg T \wedge \forall x\, Q(x)$$

i.e., the teacher is absent and all the students keep quiet.

(*b*) The even statement is:

$$\forall x\, Q(x) \wedge T$$

$\therefore$ The negation of the given statement is

$$\neg (\forall x\, Q(x) \wedge T) \equiv \neg \forall x\, Q(x) \vee \neg T$$
$$\equiv \exists x \neg Q(x) \vee \neg T$$

i.e., some students do not keep quiet or the teacher is absent.

(*c*) The given statement is:

$$\exists x \neg Q(x) \vee \neg T \equiv \neg \forall x\, Q(x) \vee \neg T$$

$\therefore$ The negation of the given statement is

$$\neg (\neg \forall x\, Q(x) \vee \neg T) \equiv \forall x\, Q(x) \wedge T$$

i.e., All the students keep quiet and the teacher is present.

(*d*) Let $D(x, y)$ represent "x has done problem y".

The given statement is

$$(\neg \exists x)\ (\forall y\, D(x, y)) \qquad \ldots(1)$$

$\therefore$ The negation of the given statement is

$$(\neg \neg \exists\ (x))\ (\forall y\, D(x, y))$$
$$\equiv \exists x\ \forall y\, D(x, y) \qquad \ldots(2)$$

i.e., some one has done every problem in the exercise.

Aliter:

(1) can be re-written as

$$\forall x\ \neg \forall y\, D(x, y)$$
$$\equiv \forall x\ \exists y,\ \neg D\ (x, y)$$

$\therefore$ The negative of this statement is

$$\neg \forall x\ \exists y \neg D(x, y) \equiv \exists x\ \neg \exists y\ \neg D(x, y)$$
$$\equiv \exists x\ \forall y\, D(x, y),$$

which is the same as (2). ■

Example 9

Show, by indirect method of proof, that $\forall x\ (p(x) \vee q(x)) \Rightarrow (\forall x\ p(x)) \vee (\exists x\ q(x))$.

Solution

Let us assume that $\neg[(\forall x\ p(x)) \vee (\exists x\ q(x))]$ as an additional premise and prove a contradiction.

Step No.	Statement	Reason
[1]	$\neg[(\forall x\ p(x)) \vee (\exists x\ q(x))]$	P(additional)
[2]	$\neg(\forall x\ p(x)) \wedge (\exists x\ q(x))$	T, 1, De Morgan's law
[3]	$\neg(\forall x\ p(x))$	T, 2, simplification
[4]	$\neg(\exists x\ q(x))$	T, 2, simplification
[5]	$\exists x\ \neg p(x)$	T, 3 and negation
[6]	$\forall x\ \neg q(x)$	T, 4 and negation
[7]	$\neg p(a)$	ES and 5
[8]	$\neg q(a)$	US and 6
[9]	$\neg p(a) \wedge \neg q(a)$	T, 7, 8 and conjunction
[10]	$\neg(p(a) \vee q(a)$	T, 9 and De Morgan's law
[11]	$\forall x\ (p(x) \vee q(x))$	P
[12]	$p(a) \vee q(a)$	US and 11
[13]	$(p(a) \vee q(a) \wedge \neg(p(a) \wedge q(a)$	T, 10, 12 and conjunction
[14]	F	T, 13

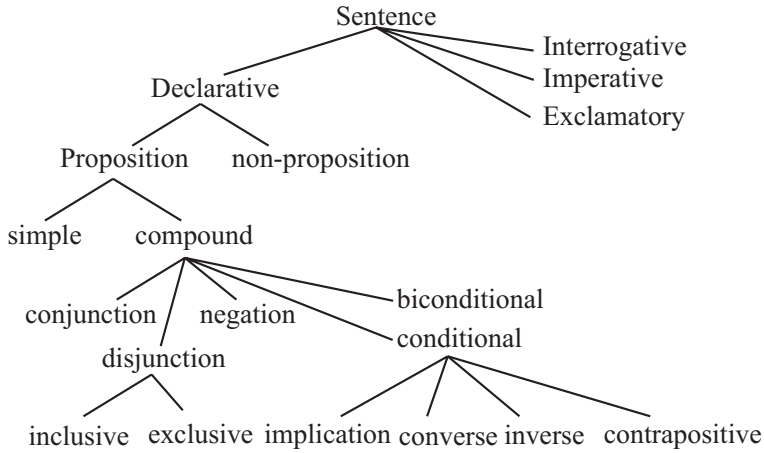
SUMMARY

1. **Proposition**
   - A proposition is a declarative sentence that is either true or false, but not both.
   - A **compound proposition** can be formed by combining tow or more simple propositions, using logical operators: $\vee$, $\wedge$, $\sim$, $\rightarrow$, and $\leftrightarrow$.
   - The conjunction of two propositions is true if and only if both components are true; their **disjunction** is true if at least one components is true. An implication is false only if the premise is true and the conclusion is false. A **biconditional** is true if and only if both compounds have the same truth value.

- The truth tables for the various logical operations can be combined into a single table, as in Table below.

p	q	$p \wedge q$	$(q \vee q)$	$\sim p$	$p \rightarrow q$	$p \leftrightarrow q$
T	T	T	T	F	T	T
T	F	F	T	F	F	F
F	T	F	T	T	T	F
F	F	F	T	T	T	T

- There new implications can be constructed from a given implication: **converse, inverse,** and **contrapositive.**
- Various types of sentences and propositions can be summarized in a tree diagram, as in Figure below:



- A **tautology** is a compound statement that is always true. A **contra diction** is a compound statement that is always false. A **contingency** is a proposition that is neither a tautology nor a contradiction.
- Two compound propositions p and q are **logically equivalent** if they have identical truth values, symbolized by $p \equiv q$

2. Argument

- An argument $H_1 \wedge H_2 \wedge \ldots \wedge H_n \rightarrow C$ is **valid** if the implication is a tautology; otherwise, it is **invalid.**

3. Quantifiers

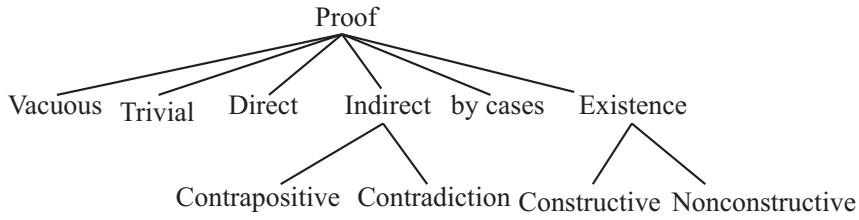
- There are two quantifiers: **universal quantifier** ($\exists$).
- A **predicate** $P(x)$ is a sentence about the properties of the object x. The set of the values of x is the **universe of discourse** (UD).

- **De Morgan's laws:**

$$\sim[(\forall x)\, P(x)] \equiv (\exists x)\, [\sim P(x)]$$
$$\sim[(\exists x)\, P(x)] \equiv (\forall x)\, [\sim P(x)]$$

4. Proof Methods

- There are six commonly used proof techniques: **vacuous proof, trivial proof, direct method, indirect method, proof by cases** and **existence proof.**



- In a direct proof, assume the given hypotheses are true. Then try to reach the given conclusion logically.
- For indirect, proof by contrapositive, assume the given conclusion is false. Then establish directly that the given hypothesis is also false.
- For indirect proof by contradiction, assume the given hypothesis is true, but the given conclusion is false. Then try to reach a contradiction.
- For a constructive existence proof of a theorem $(\exists x)\ P(x)$, produce an element b such that $P(b)$ is true. In a nonconstructive existence proof, establish the existence of such an element b by an indirect method.

5. Counter example

- To disprove the proposition $(\forall x)\ P(x)$, it suffices to produce an object c for which $P(c)$ is false.

EXERCISE

1. If $p \Rightarrow q$ is false, can you determine the truth value of $\urcorner (P \wedge Q) \Rightarrow q$? Explain.

2. Determine whether the following are tautologies:

(*a*) $p \vee [\neg (P \wedge Q)]$ (*b*) $(p \vee q) \vee p$

(*c*) $p \Rightarrow (p \wedge q)$ (*d*) $p \Rightarrow (p \vee q)$.

[**Ans.** (*a*) Yes, (*b*) Yes, (*c*) No (*d*) Yes.]

3. Prove that the following are tautologies.

(*a*) $[(p \to q) \wedge (p \vee\to r)] \Leftrightarrow [p \wedge r)]$

(*b*) $(p \wedge q) \vee [(\neg p) \wedge q] \vee p$.

(*c*) $[(p \Rightarrow r) \wedge (q - r)] \Rightarrow [(p \vee q) \Rightarrow r]$

(*d*) $\{[p \vee q) \Rightarrow r] \wedge (\neg p)\} \Rightarrow (p \Rightarrow r)$.

4. Show that

$$[\neg p \wedge (\neg q \wedge r)] \vee (q \wedge r) \vee (p \wedge r) \equiv r.$$

5. Find the disjunctive normal form of $(p \vee q \vee \neg q) \wedge (p \vee \neg p)$.

6. Find the conjunctive normal form of $q \vee (p \wedge \neg q) \vee (\neg p \wedge \neg q)$.

7. Prove that following:

(*a*) $p \vee q \Rightarrow p$ (*b*) $p \Leftrightarrow q \Rightarrow q \Rightarrow p$

(*c*) $q \Rightarrow p \to q$ (*d*) $p \wedge (q \vee r) \Rightarrow (p \wedge q) \vee r$.

8. Write down the negation of $\{\forall x, p(x)\} \vee \{\forall x, \neg q(x)\}$.

[**Ans.** $\{\exists x, \neg p(x)\}$ *n* $\{\exists x, q(x)\}$]

9. Prove that there exists a real number x such that $x^3 + 2x^2 - 5x - 6 = 0$.

10. Prove that the sum of two prime number, each larger than 2, is not a prime number.

11. Prove that n^2 is even if and only if n is even.

12. Let $A_1, A_2, A_3, ..., A_n$ and B be any sets, for $x \geq 2$. Prove that

(*a*) $\left\{\bigcap_{i=1}^{n} A_i\right\} = \bigcap_{i=1}^{n} \overline{A_i}$

(*b*) $\left\{\bigcup_{i=1}^{n} A_i\right\} \cap B = \bigcup_{i=1}^{n} (A_i \cap B)$

(*c*) $\left(\bigcap_{i=1}^{n} A_i\right) UB = \bigcap_{i=1}^{n} (A_i \, UB)$

13. If P and R are true and Q, S are false then find the truth values of $[P \wedge \neg Q \vee R] \vee [R \wedge (\neg P \Rightarrow Q)]$

[**Ans.** $R \wedge (\neg P \Rightarrow Q)$ is true $\neg (R \wedge (\neg P \Rightarrow Q)]$ is false]

14. Write the duals of

(a) $(P \vee Q) \wedge R$ (b) $(P \wedge Q) \vee T$

(c) $\neg(P \vee Q) \wedge (P \vee \neg(Q \wedge \neg S))$

[**Ans.** (a) $(P \wedge Q) \vee R$ (b) $(P \vee Q) \wedge F$ (c) $\neg(P \wedge Q) \vee (P \wedge \neg(Q \vee \neg S)$.

15. If A and A^* be dual formulae and let $P_1, P_2, ..., P_n$ be all the all the atomic variables occur in A and A^*. Prove this by implications.

16. Show that

$(x)\,(P(x) \to Q(x)) \wedge (x)\,(Q(x) \to R(x) \Rightarrow (x)\,(P(x) \to R(x))$

17. Show that $(\exists x)\,M(x)$ follows logically from the premises $(x)\,(H(x) \to M(x)$ and $(\exists x)\,H(x)$.

18. Show that $(x)\,(P(x) \vee Q(x) \Rightarrow (x)\,P(x) \vee (\exists x)\,Q(x)$.

19. Show that $\neg P(a, b)$ follows logically form $(x)\,(y)\,P(x, y) \to W\,(x, y)$ and $\neg W(a, b)$.

20. Use the indirect method to prove that the conclusion $\exists z\,Q(z)$ follows from the premises $\forall x\,(P(x) \to Q(x)$ and $\exists y\,P(y)$.

Suggested Readings

1. L. Corroll, *Symbolic logic and the same of logic,* Dover, New York, 1958.

2. N. Falletta, the Paradoxicon, Wiley, New York 1990.

3. J.T. Johnson, "*Fuzzy Logic*", *Popular Science.* Vol. 257, pp. 87–89 (Jul 1990).

4. R.M. Smullyan. *Alice in Puzzle-land,* Penguin, New York, 1984.

5. D. Solow, How to Road and Proofs, Wiley, New York, 1982.

❑❑❑

4

Relations

Karl Friedrich Gauss (1777 – 1855)

Karl Friedrich Gauss was a son of a laborer born in Brunswick, Germany. At the age of 3, he detected an error in his father's bookkeeping. The Duke of Brunswick, recognized his remarkable talent and sponsored his education. Gauss received his doctorate in 1799 from the university of Helmstedt. In his doctoral thesis, he gave the first rigorous proof of the Fundamental theorem of Algebra, which states ``Any polynomial of degree n (≥ 1) with real coefficients has at least one zero''. Although Newton and Euler, among the other brillient minds, had attempted to prove it, but faild.

Gauss made significant contributions to algbera, number theory, geometry, analysis, physics and astronomy. His work Disquisitiones Arithmeticae of 1801 laid the foundation of modern number theory. In 1807 Gauss joind University of Göttingen as a professor, director of the observatory and served till death. He was called as the ``prince of mathematics'' by his contemporary mathematicians. He made the famous statement ``Mathematics is the queen of sciences and the theory of numbers the queen of mathematics.'' The Mathematics community will be indebtable for his services.

4.1 Introduction

According to G.B. Mathews 'the invention of the symbol $\equiv$ by Gauss offered a striking example of the advantage which may be derived from an appropriate notation, and marks an apoch in the development of the science of arithmetic. The functions are a special care of rotation, which are also used is everyday life. Relations have applications to many disciplines including biology, computer science, psychology and sociology.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_4

The word 'relation' suggests some familiar examples of relations such as the human relation like father to son, mother to son, brother to sister etc. The familian examples is the arithmatic are relations such as 'greater than', 'less than' or 'equal to' between two real numbers.

Relation has got tremendous applications in almost every sphere of fields, social, economy, engineering, technology etc. In computer science and electronics the concept of relation is a major tool to learn, to understand it more clearly.

4.2 Product Sets

An ordered pair (a, b) is said to be a listing of the objects a and b in a prescribed order, with a appears first and b second. An ordered pair is merely a sequence of length 2. In sequence it follows that the ordered pair (a_1, b_1) and (a_2, b_2) are said to be equal if and only if $a_1 = a_2$ and $b_1 = b_2$.

If A and B are two nonempty sets, we may define the product set or cartesian product $A \times B$ as the set of all ordered pairs (a, b) with $a \in A$ and $b \in B$. Thus one may write

$A \times B = \{(a, b) \forall\ a \in A \text{ and } b \in B\}$.

we may define the cartesian product of three or more nonempty sets generalizing the earlier definition of the cartesian product of two sets

i.e. the cartesian product of $A_1 \times A_2 \times .... \times A_n$ of the nonempty sets $A_1, A_2, \ldots, A_m$ is the set of all ordered m – tuples $(a_1, a_{21}, \ldots, a_m)$ where $ai \in A_i$, $i = 1, 2, \ldots m$.

Thus

$A_1 \times A_2 \times \ldots. A_m = \{(a_1, a_2, \ldots, a_m) \forall\ a_i \in A_i,\ i = 1, 2, \ldots, m.$

4.3 Partitions

A partition or quotient set of a nonempty set A is a collection of nonempty subsets of A such that

(i) each element of A belongs to one of the sets in p

(ii) if A_1 and A_2 are distinct elements of p then $A_1 \wedge A_2 = \phi$

The sets in p are called the *blocks* or *calls* of the partition. As shown in Fig. 4.1. a partition $\mathrm{p} = \{ A_1, A_2, A_3, A_4, A_5, A_6, A_7,\}$ of A into seven blocks

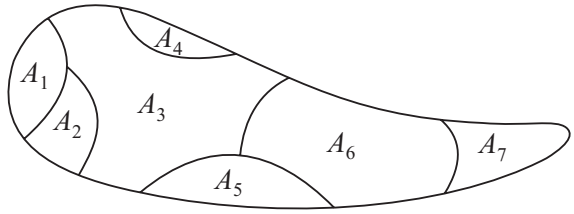


Fig. 4.1: Partitions

The Number of Partitions of a Finite set: There is a formula for computing the numbers of partition of a set with size *n.*

It is given by $\sum_{r=1}^{n} S(n_1\, r)$, where $S\,(n_1\, r)$ denotes a stirling number of the second kind, defined by

$$S\,(n,\, 1) = 1 = S\,(n,\, n)$$

$$S\,(n,\, r) = S\,(n-1,\, r-1) + r\,S\,(n-1,\, r), \qquad 1 < r < n.$$

4.4 Relations

We consider A and B as non-empty sets. A relation $\Re$ from A to B in a subset of $A \times B$. If $\Re \subseteq A \times B$ and $(x, y) \in \Re$ we can say that *x is related to y* by $\Re$ and we also write ${}_x\Re_y$. If *x is not related to y* by $\Re$, we may write ${}_a\not\Re_b$

If A and B are equal then, we say that $\Re \subseteq A \times A$ is a relation on A, instead of a relation from A to A.

Definition (Operations on Relations): Let $\Re$ and S be the relations from a set A to set B. Than $\Re \cup \mathbb{S}$, $\Re \cap \mathbb{S}$, $\Re - \mathbb{S}$, $\mathbb{S} - \Re$, and $\sim \Re$ are the relations given by

$$a(R \cup S)b \text{ if } {}_aR_b \text{ or } {}_aS_b$$

$$a(R \cap S)b \text{ if } {}_aR_b \text{ and } {}_aS_b$$

$$a(R - S)b \text{ if } {}_aR_b \text{ and } {}_a\not\Re_b$$

$$a(S - R)b \text{ if } {}_aR_b \text{ and } {}_aS_b$$

$$a(\sim R)b \text{ if } {}_a\not\Re_b$$

4.5 Binary Relations in a Set

A binary relation $\Re$ is to be defined in a set A if for any ordered pair $(x, y) \in A \times A$, is ${}_x\Re_y$ true or false.

i.e., $\Re = \{(x, y) \in A \times A : {}_x\Re_y \text{ is true}\}$.

i.e. A relation $\Re$ is a set A is a subset of $A \times A$.

Note: The meaning of binary is two. So, the binary relation is a relation between two sets, there sets may be different or identical, for the sake of convenience a binary relation will be written as only relation.

4.6 Domain and Range of a Relation

The domain Dom, of a relation $\Re$ is defined as the set of all first elements of the ordered pairs which belong to $\Re$,

i.e. $$\text{Dom.} = \{x: (x, y) \in \mathfrak{R}, \forall\, y \in A\}.$$

The range Ran, of a relation is defined as the set of all second elements of the ordered pairs which belong to R,

i.e. $$\text{Ran.} = \{y: (x, y) \in \mathfrak{R}, \forall\, y \in B\}.$$

Obviously, Dom. $\subseteq A$ and Ran. $\subseteq B$.

For example: Let $A = \{1, 2, 3, 4\}$ and $B = \{a, b, c\}$.

Every subset of $A \times B$ is a relation from A to B. So if $\Re = \{(2, a), (4, a), (4, c)\}$, then the domain of $\Re$ is the set $\{2, 4\}$ and the range of $\Re$ is the set $\{a, c\}$.

If $\Re$ is a relation from A to B and $x \in A$, we define $\Re(x)$, the $\Re$- relative set of x, to be the set of all y in B with property that x is $\Re$-related to y. Thus, in symbols.

$$\Re(x), = \{y \in B\ \forall {}_x\Re_y\}.$$

Similarly, if $A_1 \subseteq A$, then $\Re\,(A_1)$, the $\Re$-relative set of A_i, is the set of all y in B with the property that x is $\Re$-related to y for some x in A_1

i.e. $$\Re(A_1) = \{y \in B\ \forall_x R_y\ \forall\, x \in A_1\}.$$

From the preceding definitions, we can see that $\Re(A_1)$ is the union of the sets $\Re(x)$, where $x \in A_1$. The sets $\Re(x)$ play an important role in the study of many types of relations.

For example: Let $A = \{a, b, c, d\}$ and $\Re = \{(a, a), (a, b), (b, c)\ (c, a)\ (d, c), (c, b)\}$.

then $$\Re(a) = \{a, b\}, \Re(b) = \{c\} \text{ and if } A_1 = \{c, d\}$$

then $$\Re(A_1) = \{a, b, c\}.$$

4.6.1 Number of Distinct Relation From set *A* to *B*

Let the number of elements of A and B be m and n respectively. Then the number of elements of $A \times B$ is mn. Therefore, the number of elements is a power set of $A \times B = 2^{mn}$ Hence $A \times B$ has 2^{mn} different subsets.

The every subset of $A \times B$ is a relation from A to B. Hence the number of different relations A to B is 2^{mn}.

Theorem 1: For any two finite nonempty sets A and B,

Prove that $|A \times B| = |A|\ |\mathrm{B}|$

Proof: Let $|A| = m$ and $|B| = n$.

To form an ordered pair (a, b) $\forall\ a \in A, b \in B$. We have to perform two task successfully:

(i) to choose a first element from A;

(ii) to choose a second element from B;

There are m ways to perform task 1 and n ways to task 2 so, by multiplication principle, we have $m \times n$ ways to form an ordered pair (a, b) *i.e.* $|A \times B| = m.\ n = |A|\ |B|$ **Hence proved.**

Theorem 2: Let $\Re$ be the relation from A to B, and let A_1 and A_2 be subsets of A, then prove that

(i) If $A_1 \subseteq A_2$, then $\Re(A_1) \subseteq \Re(A_2)$.

(ii) $\Re(A_1 \cup A_2) = \Re(A_1) \cup \Re(A_2)$

(iii) $\Re(A_1 \cap A_2) \subseteq \Re(A_1) \cup \Re(A_2)$

Prove that equality its does not hold in case (iii) always.

Proof: (i) If $y \in \Re(A_1)$, then ${}_x\Re_y\ \forall\ x \in A_1$. Since $A_1 \subseteq A_2$, $x \in A_2$. Thus $y \in \Re(A_2)$, which proves (i)

(ii) If $y \in \Re\ (A_1 \cup \mathrm{A}_2)$, then by definition ${}_x\Re_y$ for some x in $A_1 \cup A_2$. If x is in A_1, then, since ${}_x\Re_y$ we must have $y \in \Re(A_1)$. By some argument, if x is in A_2, then $y \in \Re(A_2)$. One either case, $y \in \Re(A_1) \cup \Re(A_2)$. Thus we have shown that $\Re(A_1 \cup A_2) \subseteq \Re(A_1) \cup \Re(A_2)$.

Conversely, since $A_1 \subseteq (A_1 \cup A_2)$.

Part (i) tells us that $\Re\ (A_1) \subseteq \Re(A_1 \cup A_2)$.

Similarly, $\Re(A_2) \subseteq \Re(A_1 \cup A_2)$.

Thus $\Re(A_1) \cup \Re(A_2) \subseteq \Re(A_1 \cup A_2)$. Which proves the result.

(*iii*) If $y \in \Re(A_1 \cap A_2)$, then, for some x in $A_1 \cap A_2$, ${}_x\Re_y$. Since x is in both A_1 and A_2, it follows that y is in both $\Re(A_1)$ and $\Re(A_2)$;

i.e. $y \in \Re(A_1) \cap \Re(A_2)$.

Note: Part (*iii*) does not claim equality of sets

Let $A = Z$, $\Re$ be "$\leq$", $A_1 = \{0, 1, 2\}$, and $A_2 = \{9, 13\}$. Then $\Re(A_1)$ consists of all integers n such that $0 \leq n$, or $1 \leq n$ or $2 \leq n$.

Thus $\Re(A_1) = \{0, 1, 2,....\}$.

Similarly, $\Re(A_2) = \{9, 10, 11,....\}$

so $\Re(A_1) \cap \Re(A_2) = \{9, 10, 11,....\}$.

On the other hand $A_1 \cup A_2 = \phi$;

Thus $\Re(A_1 \cap A_2) = \phi$

This shows that the containment in the above theorem is not always in equality.

Let $A = \{1, 2, 3\}$ and $B = \{x, y, z, w, p, q\}$.

Consider the relation

$$\Re = \{(1, x), (1, z,), (2, w), (2, p), (2, q), (3, y)\}.$$

Let $A_1 = \{(1, 2)\}$ and $A_2 = \{2, 3\}$.

Then $\Re(A_1) = \{x, z, w, p, q\}$ and $\Re(A_2) = \{w, p, q, y\}$.

Thus $\Re(A_1) \cup \Re(A_2) = B$.

Since $A_1 \cup A_2 = A$.

We see that $\Re(A_1 \cup A_2) = \Re(A) = B$.

Also, $\Re(A_1) \cap \Re(A_2) = \{w, p, q\} = \Re\{(2)\} = \Re\{A_1 \cap A_2\}$

So in this case equality does hold for the containment in theorem above. ■

Theorem 3: Let $\Re$ and $\mathbb{S}$ be relation from A to B. If $\Re(a) = \mathbb{S}(a)$ for all a in A, then $\Re = \mathbb{S}$.

Proof: If ${}_a\Re_b$ than $b \in \Re(a)$

Therefore, $b \in \mathbb{S}(a)$ and $a \mathbb{S} b$.

A completely similar argument shows that, if $a \mathbb{S} b$, then $a \Re b$.

Thus $\Re = \mathbb{S}$. ■

SOLVED EXAMPLES

Example 1

Let $A = \{1, 2, 3, 4\}$ and $B = \{a, b, c\}$; $\mathcal{R} = \{(1, a), (2. b), (3, c), (4, a)\}$ and $\mathcal{S} = \{(1, b), (2, c), (3, a) (4, a)\}$. Find the value of $\mathcal{R} \cup \mathcal{S}$, $\mathcal{R} \cap \mathcal{S}$, $\mathcal{R} - \mathcal{S}$, $\mathcal{S} - \mathcal{R}$, $\sim \mathcal{R}$,$\sim \mathcal{S}$ and $\mathcal{R} \cup (\sim \mathcal{S})$

Solution

$$\mathcal{R} \cup \mathcal{S} = \{(1, a), (2, b), (3, c), (4, a), (1, b), (2, c), (3, a)\}$$

$$\mathcal{R} \cap \mathcal{S} = \{(4, a)\}$$

$$\mathcal{R} - \mathcal{S} = \{(1, a), (2, b), (3, c)\}.$$

$$\mathcal{S} - \mathcal{R} = \{(1, b), (2, c), (3, a)\}.$$

$$\sim \mathcal{R} = \{(1, b); (1, c), (2, a), (2, c), (3, b), (3, b), (4, b), (4, c)\}.$$

$$\sim \mathcal{S} = \{(1, a), (1, c), (2, a), (2, b), (3, b), (3, c), (4, b), (4, c)\}.$$

$$\mathcal{R} \cup (\sim \mathcal{S}) = \{(1, a), (1, c), (2, a), (2, b), (3, b), (3, c), (4, a), (4, b), (4, c)\}.$$

■

Example 2

For any three sets A, B, C prove that.

(i) $A \times (B \cup C) = (A \times B) \cup (A \times C)$

(ii) $(A - B) \times C = A \times C - B \times C.$

Solution

(*i*) $$\begin{aligned} A \times (B \times C) &\equiv \{(x, y) \forall\, x \in B \cup C\} \\ &\equiv \{(x, y) \forall x \in A \text{ and } y \in B \text{ or } y \in C\} \\ &\equiv \{(x, y) \forall\, (x \in A, y \in B) \text{ or } (x \in A, y \in C)\} \\ &\equiv (A \times B) \cup (A \times C) \end{aligned}$$

(*ii*) $$\begin{aligned} (A - B) \times C &\equiv \{(x, y) \forall\, x \in A - B, y \in C\}. \\ &\equiv \{(x, y) \forall\, (x \in A, x \notin B), y \in C\}. \\ &\equiv \{(x, y) \forall\, (x \in A, y \in C) \text{ and } (x \notin B, y \in C)\}. \\ &\equiv \{(x, y) \forall\, (x, y) \in A \times C, (x, \text{y}) \notin (B \times C)\}. \\ &\equiv A \times C - B \times C \end{aligned}$$

■

Example 3

For any nonempty sets A, B, C. Prove that:

(i) $A \times (B \cup C) = (A \times B) \cup (A \times C)$

(ii) $A \times (B \cap C) = (A \times B) \cap (A \times C)$

Solution

(*i*) Let $(x, y) \in A \times (B \cup C)\ \forall\ x \in A, y \in B \cup C.$

or $x \in A$ and $y \in B$ or $y \in C.$

This means that.

$x \in A, y \in B$ or $x \in A, y \in C.$

Hence $(x, y) \in A \times B$ or $(x, y) \in A \times C.$

i.e. $(x, y) \in (A \times B) \cup (A \times C).$

Which proves that

$A \times (B \cup C) \subseteq (A \times B) \cup (A \times C)$

Conversely $(x, y) \in (A \times B) \cup (A \times C)$

then $(x, y) \in (A \times B)$ or $(x, y) \in A \times C$ *i.e.* $x \in A$ and $y \in B$

or $x \in A, \quad y \in C.$

i.e. $x \in A \quad$ and $\quad y \in B$ or $y = C.$

i.e. $x \in A, \quad y \in B \cup C.$

Hence $(x, y) \in A \times (B \cup C).$

i.e. $(A \times B) \cup (A \times C) \subseteq A \times (B \cup C)$

From the above two results it follows that

$A \times (B \cup C) = (A \times B) \cup (\mathrm{A} \times C)$

(*ii*) Let $(x, y) \in A \times (B \cap C)$

Then $x \in A, y \in B \cap C$

i.e. $x \in A, y \in B$ and $y \in C.$

This means that $x \in B$ or $x \in A, y \in C.$

i.e. $(x, y) \in A \times B$ and $(x, y) \in A \times C$

so that $(x, y) \in (A \times B) \cap (A \times C)$

which proves that

$A \times (B \cap C) \subseteq (A \times B) \cap (A \times C)$

Conversely $(x, y) \in (A \times B) \cap (A \times C)$

Then $(x, y) \in A \times B$ and $(x, y) \in A \times C$

Hence $x \in A$, $y \in B$ and $x \in A$, $y \in C$.

This implies that $x \in A$, $y \in B$ and $y \in C$.

i.e. $x \in A, y \in B \cap C$

Thus $(x, y) \in A \times (B \cap C)$.

i.e. $(A \times B) \cap (A \times C) \subseteq A \times (B \cap C)$

From the above results, it follows that

$$A \times (B \cap C) = (A \times B) \cap (A \times C).$$ ■

Example 4

Let A = {1, 2, 3, ..., 10} and $\Re$ *and* $\mathbb{S}$ *be binary relations in A defined by*

$${}_m\Re_n \text{ if } m - n \text{ is divisible by 3}$$

$${}_m\mathbb{S}_n \text{ if } m - n \text{ is divisible y 4.}$$

Find $\Re \cup \mathbb{S}$, $\Re \cap \mathbb{S}$, $(\sim \Re) - \mathbb{S}$ *and* $\sim (\Re \cup \mathbb{S})$.

Solution

Defining $\Re \cup \mathbb{S}$ and $\Re \cap \mathbb{S}$ by using.

${}_m(\Re \cup \mathbb{S})_n$ if $m - n$ is divisible by 3 or 4.

${}_m(\Re \cap \mathbb{S})_n$ if $m - n$ is divisible by 3 and 4 *i.e.* 12

so $\Re \cap \mathbb{S} = \{(1, 1); (2, 2), ...(10, 10)\}$.

${}_m((\sim \Re) - \mathbb{S})_n$ if ${}_m(\sim \Re)_n$ and ${}_m(\sim \mathbb{S})_n$

${}_m((\sim \Re) - \mathbb{S})_n$ if m is neither divisible by 3 nor by 4

${}_m(\sim(\Re \cup \mathbb{S}))_n$ if $m - n$ is not divisible by 3 and not by 4.

So $\sim \Re - \mathbb{S} = \sim (\Re \cup \mathbb{S})$

If $\Re$ is a relation from A to B and S is a relation from B to C. We may combine the relation $\Re$ and $\mathbb{S}$ to get relation from A to C. ■

Note: This is called the *composition* of $\Re$ and $\mathbb{S}$.

4.6.2 Solution sets and Graph of Relations

If $\Re$ be *a* relation. The solution set $\Re^*$ of the relation $\Re$ consists of the elements (a, b) of $A \times B$ for which ${}_a\Re_b$ is true. *i.e.* $\Re^* = \{(a, b): a \in A, b \in B, {}_a\Re_b \text{ is true.}\}$.

Note: (*i*) $\Re^*$ the solution of a relation $\Re$ from A to B is a subset of $A \times B$.

(*ii*) $\Re^*$ can be representad, *i.e.* plotted on the coordinate diagram of $A \times B$.

(*iii*) $A \times B$ belong to the solution set of $\Re$.

4.6.3 Relation as Sets of Ordered Pairs

If $\Re^*$ is any subset of $A \times B$. We define a relation $\Re$ where ${}_x\Re_y$ ready $(x, y) \in \Re^*$

The solution set of this relation $\Re$ is the original set $\Re^*$

To every relation $\Re$ there corresponds a unique solution set $\Re^* \leq A \times B$ and to every subset of $\Re^*$ of $A \times B$ there corresponds a relation $\Re$ for which $\Re^*$ is its solution set.

4.7 The Matrix of a Relation and Digraphs

We can represent a relation between two finite sets with a matrix as follows:

If $A = \{a_1, a_2, ..., a_m\}$ and $B = \{b_1, b_2, ..., b_m\}$ are finite sets containing m and n elements, respectively, and $\Re$ is a relation from A to B, we represent $\Re$ by the $m \times$ n matrix $M_R = [m_{ij}]$, which is defined by

$$m_{ij} = \begin{cases} 1 & \text{if } (a_i, b_j) \in \Re \\ 0 & \text{if } (a_i, b_j) \in \Re \end{cases}$$

The matrix M_R is called the matrix of $\Re$. Often M_R provides an easy way to check whether $\Re$ has a given property.

For example. If we consider the matrix $M = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix}$ since M is 3×4, we assume

$$A = \{a_1, a_2, a_3\} \text{ and } B = \{b_1, b_2, b_3, b_4\}.$$

Then $(a, b_j) \in \Re$ iff $m_{ij} = 1$.

Thus

$$\Re = \{(a_1, b_1), (a_1, b_4), (a_2, b_2), (a_2, b_3), (a_3, b_1), (a_3, b_3)\}.$$

If A is a finite set and $\Re$ is a relation on A, we can also represent $\Re$ pictorially as follows. We draw a small circle for each element of A and label the circle with the corresponding elements of A. These circle are called vertices. We also draw an arrow, called an edge, from vertex a_i to vertex a_j iff $a_i \Re a_j$. The resulting pictorial representation of $\Re$ is called a directed graph or digraph of $\Re$.

Relations R on a finite set A can be represented pictorially in yet another way. We denote every element of A by a point, called a **Vertex** (or **node**), and each ordered pair (a, b) in R by a directed arc or a directed line segment, called an **edge,** from a to b. The resulting diagram is a **directed graph** or simply a **digraph.** If an edge (a, b) exists, we say that vertex b is **adjacent** to vertex a. (Notice the order of the vertices.)

Thus if $\mathcal{R}$ is a relation on A, the edgel in the digraph of $\mathcal{R}$ correspond exactly to the pairs in $\mathcal{R}$, and the vertices correspond exactly to the elements of the set A. Sometimes, when we want to emphasize the geometrical nature of some property of $\mathcal{R}$, we may refer to the pairs of $\mathcal{R}$ themselves as edge and the elements of A as vertices.

Let $A = \{1, 2, 3, 4\}$ and

$\mathcal{R} = \{(1, 1), (1, 2), (2, 1), (2, 2), (2, 3), (2, 4), (3, 4), (4, 1)\}$.

Then the digraph of $\mathcal{R}$ is as shown below (Fig. 4. 2).

An important concept for relations is inspired by the visual form of digraph. If $\mathcal{R}$ is a relation on a set A and $a \in A$, then the **in-degree** of a is the number of $b \in A$ such that $(b, a) \in \mathcal{R}$. The **out-degree** of a is the number of $b \in A$ such that $(a, b) \in \mathcal{R}$. If $\mathcal{R}$ is a relation on a set A, and B is a subset of A, the restriction of $\mathcal{R}$ to B is $\mathcal{R} \cap (B \times B)$.

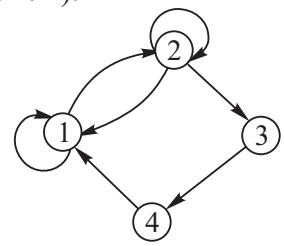


Fig. 4.2: Digraph

4.8 Paths in Relations and Digraphs

Suppose-that $\mathcal{R}$ is a relation on a set A. A path of length n in $\mathcal{R}$ from a to b is a finite sequence $\pi : a, x_1, x_2...., x_{n-1}, b$, beginning with a and ending with b, such that ${}_a\mathcal{R}_{x1}, {}_{x1}\mathcal{R}_{x2}, ..., x_{n-1}\,\mathcal{R}_b$.

A path of length n involves $n + 1$ elements of A, although they are not necessarily distinct.

A path is most easily visualized with the aid of the digraph of the relation. It appears as a geometric path or succession of edges in such a digraph, where the indicated directions of the edges are followed, and in fact a path derives its name from this representation. Thus the length of a path is the number of edges in the path, where the vertices need not be distinct.

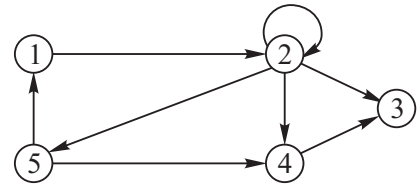


Fig. 4.3

Let us take a digraph as shown in Fig. 4.3.

π_1: 1, 2, 5, 4, 3 is a path of length 4 from vertex 1 to vertex 3.

π_2: 1, 2, 5, 1 is a path of length 3 from vertex 1 to itself and

π_3: 2, 2 is a path of length 1 from vertex 2 to itself.

A path that begins and ends at the same vertex is called a *cycle.* In above π_2 and π_3 are cycles of length 3 and 1, respectively. It is clear that the paths of length 1 can be identified with ordered pairs (x, y) that belong to $\mathcal{R}$. Paths in a relation $\mathcal{R}$ can be used to define new relations that are quite useful. If n is a fixed positive integer, we define a relation $\mathcal{R}^n$ on A as follows.

(*i*) ${}_x\mathcal{R}^n_y$ means that there is a path of length n from x to y in $\mathcal{R}$.

(*ii*) ${}_x\mathcal{R}^\infty_y$ means that there is some path in $\mathcal{R}$ from x to y.

The length of such a path will depend on x and y.

The relation $\mathcal{R}^\infty$ is sometimes called the connectivity relation for $\mathcal{R}$.

Note: (*i*) $\mathcal{R}^n(x)$ consists of all vertices that can be reached from x by means of a path in $\mathcal{R}$ of length n

(*ii*) The set $\mathcal{R}^\infty(x)$ consists of all vertices that can be reached from x by some path in $\mathcal{R}$.

(*iii*) Let R be a relation on a set A, and let $a, b \in A$. A **path** in R, that is, in the digraph of R, from a to b is a finite sequence of edges $(a, x_1), (x_1, x_2), ..., (x_{n-1}, b)$; the vertices x_i 's need *not* be distinct. The path from a to b is also denoted by $a - x_1 - x_2 - ... - x_{n-1} - b$. The number of edges in the path is its **length.** A path that begins and terminates at the same vertex is a **cycle.** A cycle of length one is a **loop.**

SOLVED EXAMPLES

Example 1

Let $A = \{a, b, c, d\}$, *and let* $\mathcal{R}$ *be the relation on A that has the matrix*

$$M_{\mathcal{R}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

Construct the digraph of $\mathcal{R}$ *and list in-degrees and out-degrees of all vertices.*

Solution

The digraph of $\mathcal{R}$ is shown in Fig. 4.4 below.

The following table gives the in-degrees and out-degrees of all vertices.

	a	*b*	*c*	*d*
in-degree	2	3	1	1
out-degree	1	1	3	2

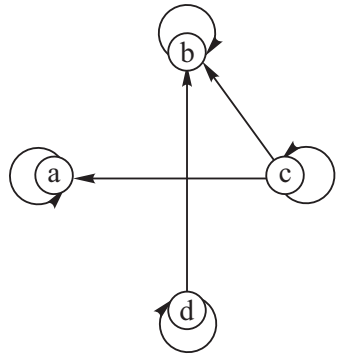


Fig. 4.4 ■

Example 2

Let A = {1, 4, 5}, and let $\mathcal{R}$ be given by the digraph shown in Fig. 4.5. Find M_R and $\mathcal{R}$.

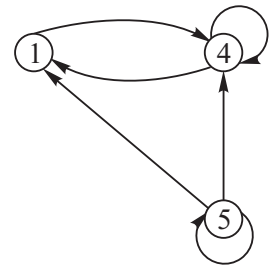


Fig. 4.5

Solution

$$M_R = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

$\mathcal{R} = \{(1, 4,), (1, 5), (4, 1), (4, 4), (5, 4), (5, 5)\}$. ■

Example 3

Let $A = \{a, b, c, d, e, f\}$ *and*

$\mathcal{R} = \{(a, a), (a, c), (b, c), (a, e), (b, e), (c, e)\}$.

Let $B = \{a, b, c\}$. *Find* $B \times B$ *and the restriction of* $\mathcal{R}$ *to* B.

Solution

$B \times B = \{(a, a), (a, b), (a, c), (b, a), (b, b), (b, c), (c, a), (c, b) (c, c)\}$.

Restriction of $\mathcal{R}$ to $B = \{(a, a), (a, c), (b, c)\}$. ■

Example 4

Let $A = \{a, b, c, d, e\}$ *and*

$\mathcal{R} = \{(a, a), (a, b), (b, c), (c, e), (c, d), (d, e)\}$.

Compute (i) $\mathcal{R}^2$ *(ii)* $\mathcal{R}^\infty$.

Solution

(*i*) The digraph of $\mathcal{R}$ is shown in Fig. 4.6.

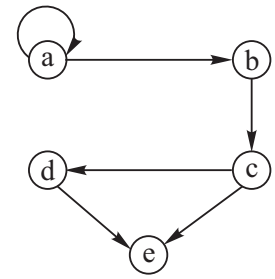


Fig. 4.6

$_a\mathcal{R}^2_a$ since $_a\mathcal{R}_a$ and $_a\mathcal{R}_a$

$_a\mathcal{R}^2_b$ since $_a\mathcal{R}_a$ and $_a\mathcal{R}_b$

$_a\mathcal{R}^2_c$ since $_a\mathcal{R}_b$ and $_b\mathcal{R}_c$

$_b\mathcal{R}^2_e$ since $_b\mathcal{R}_c$ and $_c\mathcal{R}_e$

$_b\mathcal{R}^2_d$ since $_b\mathcal{R}_c$ and $_c\mathcal{R}_d$

$_c\mathcal{R}^2_e$ since $_c\mathcal{R}_d$ and $_d\mathcal{R}_e$.

Hence $\mathcal{R}^2 = \{(a, a), (a, b), (a, c), (b, e), (b, d), (c, e)\}$.

(*ii*) To compute $\mathcal{R}^\infty$, we need all ordered pairs of vertices for which there is a path of any length from the first vertex to the second. We see that

$\mathcal{R}^\infty = \{(a, a), (a, b), (a, c), (a, d), (a, e), (b, c), (b, d), (b, e), (c, d), (c, e), (d, e)\}$.

e.g. $(a, d) \in \mathcal{R}^\infty$, since there is a path of length 3 from a to d: a, b, c, d.

Similarly, $(a, e) \in \mathcal{R}^\infty$, since there is a path of length 3 from a to e: a, b, c, e as well as a path of length 4 from a to e: a, b, c, d, e. ■

Note: Let R be a relation on a finite set $A = \{a_1, a_2, ..., a_n\}$ and let M_R be then $n \times n$ matrix representing $\mathcal{R}$. We will show how the matrix M_{R^2} of $\mathcal{R}_{R^2}$ can be computed from M_R.

4.9 Boolean Matrices

A boolean matrix is a matrix with bits as its entries. Thus $A = (a_{ij})_{m\times n}$ is a boolean matrix if $a_{ij} = 0$ or 1 for every i and j.

For instance, $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$ is a boolean matrix whereas $\begin{bmatrix} 1 & -3 \\ 0 & 4 \end{bmatrix}$ is not a boolean matrix.

4.9.1 Boolean Operations AND and OR

The boolean operations *and* ($\wedge$) and *or* ($\vee$), signal the combining of boolean matrices to construct now over. Listed below are some properties of bit operations which can be verified easily.

Let a and b be arbitrary bits. Then:

- $a \wedge a = a$
- $a \vee a = a$
- $a \wedge (b \wedge c) = (a \wedge b) \wedge c$
- $a \vee (b \wedge c) = (a \wedge b) \wedge (a \vee c)$
- $a \wedge b = b \wedge a$
- $a \vee b = b \vee a$
- $a \vee (b \vee c) = (a \vee b) \vee c$
- $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$

4.9.2 Joint and Meet

The **join** of the boolean matrices $A = (a_{ij})_{m \times n}$ and $B = (b_{ij})_{m \times n}$, denoted by $A \vee B$, is defined by $A \vee B = (a_{ij} \vee b_{ij})_{m \times n}$. Each element of $A \vee B$ is obtained by *oring* the corresponding elements of A and B. The **meet** of A and B, denoted by $A \wedge B$, is defined by $A \wedge B = (a_{ij} \wedge b_{ij})_{m \times n}$. Every element of $A \wedge B$ is obtained by *ending* the corresponding elements of A and B.

4.9.3 Boolean Product

The **boolean product** of the boolean matrices $A = (a_{ij})_{m \times p}$ and $B = (a_{ik})_{p \times n}$, denoted by $A \odot B$, is the matrix $C = (c_{ij})_{m \times n}$, where $c_{ij} = (a_{i1} \wedge b_{1j}) \vee (a_{i2} \wedge b_{2j}) \vee \ldots \vee (a_{ip} \wedge b_{pj})$.

For example,

$$\begin{bmatrix} a_{11} & a_{11} & \cdots & a_{1p} \\ \cdot & \cdot & \cdots & \cdot \\ a_{t1} & a_{i2} & \cdots & a_{ip} \\ \cdot & \cdot & \cdots & \cdot \\ a_{m1} & a_{m2} & \cdots & a_{mp} \end{bmatrix} \odot \begin{bmatrix} b_{11} & \cdots & b_{1j} & \cdots & b_{1n} \\ \cdot & \cdots & \cdot & \cdots & \cdot \\ b_{i1} & \cdots & b_{ij} & \cdots & b_{in} \\ \cdot & \cdots & \cdot & \cdots & \cdot \\ b_{p1} & \cdots & b_{pj} & \cdots & b_{pn} \end{bmatrix} = \begin{bmatrix} c_{11} & \cdots & c_{1j} & \cdots & c_{1n} \\ \cdot & \cdots & \cdot & \cdots & \cdot \\ c_{i1} & \cdots & c_{ij} & \cdots & c_{in} \\ \cdot & \cdots & \cdot & \cdots & \cdot \\ c_{m1} & \cdots & c_{mj} & \cdots & c_{mn} \end{bmatrix}$$

Notice the similarity between this definition and that of the usual product of matrices.

4.9.4 Boolean Power of a Boolean Matrix

Let A be an $m \times m$ booolean matrix and n any positive integer. The nth **boolean power** of A, denoted by $A^{[n]}$, is defined recursively as follows:

$$A^{[0]} = I_m \quad \text{(the identity matrix)}$$
$$A^{[n]} = A^{[n-1]} \odot A \text{ if } n \geq 1$$

SOLVED EXAMPLES

Example 1

If

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Find $A \vee B$ *and* $A \wedge B$.

Solution

OR operation
$$A \vee B = \begin{bmatrix} 1 \vee 0 & 0 \vee 0 & 1 \vee 1 \\ 0 \vee 1 & 1 \vee 0 & 0 \vee 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

AND operation
$$A \wedge B = \begin{bmatrix} 1 \wedge 0 & 0 \wedge 0 & 1 \wedge 1 \\ 0 \wedge 1 & 1 \wedge 0 & 0 \wedge 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$ ■

Example 2.

If

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ and } B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}$$

Find $A \odot B$ *and* $B \odot A$, *if defined.*

Solution

Since the number of columns in A equals the number of rows in B, $A \odot B$ is defined:

$$A \odot B = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}$$
$$= \begin{bmatrix} (1 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) & (1 \wedge 0) \vee (0 \wedge 1) \vee (1 \wedge 0) \\ (0 \wedge 1) \vee (1 \wedge 1) \vee (0 \wedge 0) & (0 \wedge 0) \vee (1 \wedge 1) \vee (0 \wedge 0) \end{bmatrix}$$
$$= \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

Number of columns in $B = 2$ = Number of rows in A. Therefore, $B \odot A$ is also defined:

$$B \odot A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} (1 \wedge 1) \vee (0 \wedge 0) & (1 \wedge 0) \vee (0 \wedge 1) & (1 \wedge 1) \vee (0 \wedge 0) \\ (1 \wedge 1) \vee (1 \wedge 0) & (1 \wedge 0) \vee (1 \wedge 1) & (1 \wedge 1) \vee (1 \wedge 0) \\ (0 \wedge 1) \vee (0 \wedge 0) & (0 \wedge 0) \vee (0 \wedge 1) & (0 \wedge 1) \vee (0 \wedge 0) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$ ■

Example 3

$$If\ A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Compute $A^{[2]}$ *and* $A^{[3]}$ *.*

Solution

$$A^{[2]} = A^{[1]} \odot A = A \odot A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$A^{[3]} = A^{[2]} \odot A = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$ ■

Note: We can verify that in this case, $A^{[n]} = A$ (for every $n \geq 1$.)

CTM: Let A, B, and C be three boolean matrices. Then:

- $A \vee A = A$
- $A \wedge A = A$
- $A \vee B = B \vee A$
- $A \wedge B = B \wedge A$
- $A \vee (B \vee C) = (A \vee B) \vee C$
- $A \wedge (B \wedge C) = (A \wedge B) \wedge C$
- $A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$
- $A \wedge (B \wedge C) = (A \wedge B) \vee (A \wedge C)$
- $A \odot (B \odot C) = (A \odot B) \odot C$

The sizes of the matrices are assumed compatible for the corresponding matrix operations.

4.10 Adjacency Matrix of a Relation

A relation $\Re$ from a set $\{a_1, a_2, ..., a_m\}$ to a set $\{b_1, b_2, ..., b_m\}$ can be reprinted by the $m \times n$ boolean matrix $M_R = (m_{ij})$, where

$$m_{ij} = \begin{cases} 1 & \text{if } {}_{a_i}\Re_{b_j} \\ 0 & \text{otherwise} \end{cases}.$$

M_R is the adjacency matrix of the relation R.

For exanple: Define a relation $\Re$ from A = {chicken, dog, cat} to B = {fish, rice, cotton} by $\Re = \{(a, b) | a \text{ eats } b\}$. Then R = {(chiken, fish), (chiken, rice), (dog, fish), (dog, rice), (cat, fish), (cat, rice)}. Its adjacency matrix is

$$M_R = \begin{array}{c} \\ \text{chicken} \\ \text{dog} \\ \text{cat} \end{array} \begin{array}{c} \begin{array}{ccc} \text{fish} & \text{rice} & \text{cotton} \end{array} \\ \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$$

4.11 Gray Code

Suppose a switching network is composed of n switches a_i, where $1 \le i \le n$. Let $a_i = 1$ denote that switch a_i is closed and $a_i = 0$ denote that it is open. Every state of the network can be denoted by the n-bit word $a_1a_2.... a_n$. Let $\sum^n$ denote the set of n-bit words, that is, the set of all states of the network. For example, $\sum^3$ = {000, 001, 010, 100, 011, 101, 110, 111}. Naturally, we are tempted to ask: *Is it possible to test every state of the circuit by changing the state of exactly one switch? That is, is it possible to list every n-bit word by changing exactly one bit?*

Another definition can lead to rewording the question. Two n-bit words are **adjacent** if they differ in exactly one bit, that is, if the **Hamming distance** between them is one. For example, 010 and 011 are adjacent, whereas 001 and 110 are not.

Define a relation $\Re$ on $\sum^n$ as ${}_{\alpha}\Re_{\beta}$ if α and β are adjacent. We can rephrase this: Is it possible to arrange the elements α_i of $\sum^n$ in such a way that $\alpha_i \Re \alpha_{i+1}$ where $1 \le i \le m - 1$, ${}_{\alpha_m}\Re_{\alpha_1}$, and $m = 2^n$? That is, is it possible to arrange the n-bit words around a circle in such a way that any two neighbouring words are adjacent?

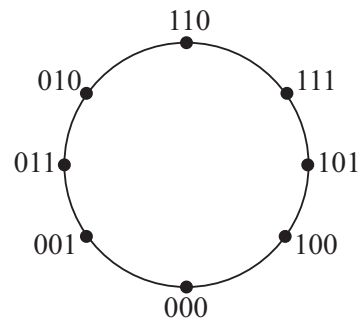


Fig. 4.7

Interestingly enough, the elements of $\sum^3$ can be arranged in this fashion 000, 001, 011, 010, 110, 111, 101, 100. Such an ordering is called a **Gray code** for $\sum^3$. More generally, a **Gray code** for $\sum^n$ is an arrangement of its elements $\alpha_1, \alpha_2,..., \alpha_m$ such that $\alpha_i \Re \alpha_{i+1}$ and $\alpha_m \Re \alpha_1$, where $1 \le i \le m - 1$. Gray codes are named for Frank Gray, who invented them in the 1940s at what was then AT&T Bell Labs.

We can restate our original question again: *Is there a Gray code for* $\sum^n$ *for every* $n \ge 1$*?* Induction leads to an affirmative answer.

Proof (by induction): Let P(n): There exists a Gray code for every $\sum^n$.

Basis step: When $n = 1$, {0, 1} is clearly a Gray code; so P(1) is true.

Induction step: Assume $P(k)$ is true; that is, there is a Gray code for $\sum^k$. Suppose $\{\alpha_1, \alpha_2,..., \alpha_r\}$ is a Gray code, where $r = 2^k$.

To show that P(k + 1) is true:

Consider the $(k + 1)$-bit words $0\alpha_1, 0\alpha_2,..., 0\alpha_r, 1\alpha_r, 1\alpha_{r-1},..., 1\alpha_1$.

Clearly they form $2r = 2^{k+1}$ elements of $\sum^{k+1}$. Call them $\beta_1, \beta_2,..., \beta_{2r}$, respectively, for convenience. Since $\alpha_i R \alpha_{i+1}$ and $\alpha_r R \alpha_1$, $\beta_j R \beta_{j+1}$ and $\beta_{2r} R \beta_1$, so $\{\beta_1, \beta_2,..., \beta_{2r}\}$ is a Gray code; that is, $P(k + 1)$ is true.

Thus, by induction, a Gray code exists for every $\sum^n$.

Theorem 4: If $\Re$ is relation on $A = \{a_1, a_2, a_3...., a_n\}$ then prove that $M_{R^2} = M_R \odot M_R$.

Proof: Let $M_R = [m_{ij}]$ and $M_{R^2} = [n_{ij}]$

By definition, the i, j^{th} element of $M_R \odot M_R$ is equal to 1. iff row i of M_R and column j of M_R have a 1 in the same relative position (k, say).

This means that $m_{ik} = 1$ and $m_{kj} = 1$ for k, $1 \le k \le n$.

By definition for the matrix M_R, the condition above means that $a_i \Re a_k$ and ${}_{a_k}\Re_{a_j}$.

Thus ${}_{a_k}\Re^2_{a_j}$ and so $n_{ij} = 1$.

We have therefore shown that position i, j of $M_R \odot M_R$ is equal to 1. iff $n_{ij} = 1$.

i.e. $M_R \odot M_{R2}$ ■

Note. (*i*) We may denote $M_R \odot M_R$ as $(M_R)^2_{\odot}$

(*ii*) $\odot$ is the symbol of boolean product of matrices.

(*iii*) We can show by the same process that

$$M_{R^3} = M_R \odot (M_R \odot M_R) = (M_R)^3_{\odot}$$

Theorem 5: For $n \geq 2$ and $\mathcal{R}$ a relation on a finite set A, we have

$$M_{R^n} = M_R \odot M_R \odot \ldots\ldots \odot M_R \ (n \text{ factors})$$

Proof: Let $P(n)$ be the assertion that the statement above holds for an integer $n \geq 2$.

Basic step: $P(2)$ is true.

Induction step: We now show that if $P(k)$ is true, then $P(k + 1)$ is true. Let us consider the matrix $M_{R^{k+1}}$.

Let $M_{R^{k+1}} = [x_{ij}]$, $M_{R^k} = [y_{ij}]$ and $M_R = [m_{ij}]$.

If $x_{ij} = 1$, we must have a path of length $(k + 1)$ from a_i to a_j.

Let a_s be the vertex that this path reaches just before the last vertex a_j, then there is a path of length k from a_i to a_s and a path of length 1 from a_s to a_j. Thus $y_{is} = 1$ and $m_{sj} = 1$.

So $M_{R^k} \odot M_R$ has a 1 in position i, j. We can see, similarly, that if $M_{R^k} \odot M_R$ has a 1 in position i, j, then $x_{ij} = 1$.

This means that $M_{R^{k+1}} = M_{R^k} \odot M_R$.

By induction

$P(k)$: $M_{R^k} = M_R \odot \ldots . \odot M_R$ (k factors) so, by substitution

$M_{R^{k+1}} = (M_R) \odot \ldots . \odot M_R \odot M_R . \mathrm{d}_{\mathrm{dd}}$

$P(k = 1) : M_R \odot \ldots . \odot M_R \odot M_R$ ($k + 1$ factors) and $P(k + 1)$ is true.

Thus by the principle of mathematical induction, $P(n)$ is true for all $n \geq 2$.

This proves the theorem. As before, we write

$M_R \odot \ldots . \odot M_R$ (n factors) as $(M_R)^n{}_{\odot}$.

4.12 Properties of Relations

The relations on finite sets can be represented by matrices, their properties can be identified from them adjacency matrices. We consider a relation R, is logically equivalent to, on the set of propositions. Since every proposition is logically equivalent to itself, it has the property that ${}_xR_x$ for every proposition x.

In many applications of computer science and applied mathematics, we deal with relations on a set A rather than relation from A to B.

4.12.1 Reflexive and Irreflexive Relations

A relation $\Re$ on a set A is *reflexive* if $(a, a) \in \Re$ for all $a \in A$, that is, if $a \Re a$ for all $a \in A$. A relation $\Re$ on a set A is *irreflexive* if $a \Re a$ for every $a \in A$.

Thus $\Re$ is reflexive if every element $a \in A$ is related to itself and it is irreflexive if no element is related to itself.

For example.

(a) Let $\Delta = \{(a, a) : a \in A$, so that Δ is the relation of *equality* on the set A. Then Δ is reflexive, since $(a, a) \in \Delta$ for all $\in A.\}$

(b) Let $\Re = \{(a, a) \in A \times A : a \neq b\}$, so that $\Re$ is the relation of *inequality* on the set A. Then $\Re$ is irreflexive, since $(a, a) \notin \Re$ for all $a \in A$.

(c) Let $A = \{1, 2, 3\}$, and let $\Re = \{(1, 1), (1, 2)\}$. Then $\Re$ is not reflexive since $(2, 2) \notin \Re$ and $(3, 3) \notin \Re$. Also $\Re$ is not irreflexive, since $(1, 1) \in \Re$.

(d) Let A be a nonempty set. Let $\Re = \varnothing \subseteq A \times A$, the *empty relation.* Then $\Re$ is not reflexive, since $(a, a) \notin \Re$ for all $a \in A$ (the empty set has no elements). However, $\Re$ is irreflexive.

$A : \Re$ is reflexive iff $\Delta \subseteq \Re$, and $\Re$ is irreflexive iff $\Delta \cap \Re = \phi$

Note: If $\Re$ is reflexive on a set A, then Dom $(\Re) = \Re an\ (\Re) = A$

4.12.2 Symmetric, Asymmetric and Antisymmetric Relations

A relation $\Re$ on a set A is *symmetric* if whenever ${}_a\Re_b$, then ${}_b\Re_a$. It then follows that $\Re$ is not symmetric if we have some a and $b \in A$ with ${}_a\Re_b$, $\Re_a$. A relation $\Re$ on a set A is *asymmetric* if whenever ${}_a\Re_b$ then ${}_b\Re_a$. It then follows that $\Re$ is not asymmetric if we have some a and $b \in A$ with both ${}_a\Re_b$ and ${}_b\Re_a$.

A relation $\Re$ on a set A is *antisymmetric* if whenever ${}_a\Re_b$ and ${}_b\Re_a$, then $a = b$. The contrapositive of this definition is that $\Re$ is antisymmetric if whenever $a \neq b$, then ${}_a\Re_b$ or ${}_b\Re_a$. It follows that $\Re$ is not antisymmetric if we have a and b in A, $a \neq b$, and both ${}_a\Re_b$ and ${}_b\Re_a$.

Given a relation $\Re$, we shall want to determine which properties hold for $\Re$. Keep in mind the following remark. A property fails to hold in general if we can find one situation where the property does not hold.

4.12.3 Transitive Relation

We say that a relation $\mathcal{R}$ on a set A is transitive if whenever ${}_a\mathcal{R}_b$ and ${}_b\mathcal{R}_c$, then ${}_a\mathcal{R}_c$. It is often convenient to say what it means for a relation to be not transitive. A relation $\mathcal{R}$ on A is not transitive if there exist a, b and c in A so that ${}_a\mathcal{R}_b$ and ${}_b\mathcal{R}_c$, but ${}_a\mathcal{R}_c$. If such a, b and c do not exist, then $\mathcal{R}$ is transitive.

SOLVED EXAMPLES

Example 1

Which of the following relations on A = {x, y, z} are reflexive?

- $R_1 = \{(x, x), (x, y), (y, y), (z, z)\}$
- $R_2 = \{(x, x), (y, y), (y, z), (z, y)\}$
- $R_3 = \varnothing$ *[the* ***empty relation****]*
- $R_4 = \{(x, x), (y, y), (z, z)\}$

Solution

For a relation $\mathcal{R}$ on A to be reflexive, every element in A must be related to itself, that is, $(a, a) \in R$ for every $a \in A$. The element a has three choices, namely. x, y, and z; therefore, the ordered pairs (x, x), (y, y), and (z, z) must be in the relation for it to be reflexive. Consequently, the relations, $\mathcal{R}_1$ and $\mathcal{R}_4$ are reflexive, whereas $\mathcal{R}_2$ and $\mathcal{R}_3$ are *not*. ■

Example 2

Find the number of reflexive relations $\mathcal{R}$ that can be defined on a set with n elements.

Solution

Since $\mathcal{R}$ is reflexive, every element on the main diagonal of M_R is 1; there are n such element. Since M_R contains n^2 elements, there are $n^2 - n = n(n - 1)$ elements that do not lie on the main diagonal; each can be a 0 or 1; so each such element m_{ij} has two choices. Thus, by the multiplication principle, we can form $2^{n(n-1)}$ such adjacency matrices, that is, $2^{n(n-1)}$ reflexive relations on A. ■

Example 3

Let $A = Z$, *the set of integers, and let*

$\mathcal{R} = \{(a, b) \in A \times A : a < b\}$

so that $\mathcal{R}$ *is the relation less than. Is* $\mathcal{R}$ *symmetric, asymmetric, or antisymmetric?*

Solution

Symmetry : If $a < b$, then it is not true that $b < a$, so $\mathcal{R}$ is not symmetric.

Asymmetry : If $a < b$, then $b \not< a$ (b is not less than a), so $\mathcal{R}$ is asymmetric.

Antisymmetry : If $a \neq b$, then either $a \not< b$ or $b \not< a$, so that $\mathcal{R}$ is antisymmetric. ■

Example. 4

Let $A = Z^+$*, the set of positive integers, and let* $\mathcal{R} = \{(a, b) \in A \times A : a$ *divides* $b\}$.

Is $\mathcal{R}$ *symmetric, asymmetric, or antisymmetric?*

Solution

If $a \mid b$, it does not follows that $b \mid a$, so $\mathcal{R}$ is not symmetric. For example, $2 \mid 3$, but $3 \mid 2$.

If $a = b = 3$, say then ${}_a\mathcal{R}_b$ and ${}_b\mathcal{R}_a$, so $\mathcal{R}$ is not asymmetric.

If $a \mid b$ and $b \mid a$, then $a = b$, so $\mathcal{R}$ is antisymmetric. ■

Matrix properties: We now relate symmetric, asymmetric, or an atisymmetric properties of a relation to properties of its matrix. The matrix $M_R = [m_{ij}]$ of a symmetric relation satisfies the property that

if $m_{ij} = 1$, then $m_{ji} = 1$.

Moreover, if $m_{ji} = 0$, then $m_{ij} = 0$. Thus M_R is a matrix such that each pair of entries, symmetrically placed about the main diagonal, are either both 0 or both 1. It follows that $M_R = M_R^T$, so that M_R is a symmetric matrix.

The matrix $M_R = [m_{ij}]$ of an asymmetric relation $\mathcal{R}$ satisfies the property that if $m_{ij} = 1$, then $m_{ji} = 0$.

If $\mathcal{R}$ is asymmetric, it follows that $m_{ii} = 0$ for all i; that is, the main diagonal of the matrix M_R consists entirely of 0, s. This must be true since the asymmetric property implies that if $m_{ii} = 1$, then $m_{ii} = 0$, which is a contradiction.

Finally, the matrix $M_R = [m_{ij}]$ of an antisymmetric relation $\Re$ satisfies the property that if $i \neq j$, then $m_{ij} = 0$.

An undirected edge between a and b, in the graph of a symmetric relation $\Re$, corresponds to a set $[a, b]$ such that $(a, b) \in \Re$ and $[b, a] \in \Re$. Sometimes we will also refer to such a set $[a, b]$ as an *undirected edge* of the relation $\Re$ an call a and b *adjacent vertices.*

Example 5

Find the number of symmetric relation that can be defined on a set with n elements.

Solution

Let R be a relation on the set and let $M_R = (m_{ij})_{n\times n}$. Then $m_{ij} = 1$ if and only in $m_{ji} = 1$ for every i and j. So each element m_{ij} below the main diagonal determines uniquely the corresponding element m_{ji} above the main diagonal; in other words, each m_{ji} has one choice (see Figure).

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ & 1 & \cdots & 1 \\ & & \ddots & \\ & & & 1 \end{bmatrix}$$

$$\begin{bmatrix} \cdot & 1 & 1 & \cdots & 1 \\ \cdot & \cdot & 1 & \cdots & 1 \\ \cdot & \cdot & \cdot & \cdot & \\ \cdot & \cdot & \cdot & \cdot & \\ & & & & 1 \\ \cdot & \cdot & \cdot & \cdots & \cdot \end{bmatrix}$$

Now, each element on or below the main diagonal has two choices: 0 or 1 (see Figure). There are $1 + 2 + ... + n = n(n + 1)/2$ such elements. So, by the multiplication principle, the number of such adjacency matrices equals $2^{n(n+1)/2}$; that is, we can define $2^{n(n+1)/2}$ symmetric relations on the set. ■

Example 6

Which of the following relations on {a, b, c} are transitive?

- $R_1 = \{(a, b), (b, c), (a, c)\}$
- $R_2 = \{(a, a), (a, b), (a, c), (b, a), (b, c)\}$
- $R_3 = \{(a, a), (b, b), (c, c)\}$
- $R_4 = \{(a, b)\}$

Solution

The relation R_1 is transitive; so are R_3 and R_4 by default. In relation R_2, $(b, a) \in R_2$ and $(a, b) \in R_2$, but $(b, b) \neq R_2$. So, R_2 is *not* transitive. ■

Example 7

Let $A = \{1, 2, 3, 4\}$ *and let*

$\Re = \{(1, 2), (1, 3), (4, 2)\}$.

Is $\Re$ *transitive?*

Solution

Since there are no elements a , b and c in A such that ${}_a\Re_b$ and ${}_b\Re_c$, but ${}_a\Re_c$, we conclude that $\Re$ is transitive. ■

Note. A relation $\Re$ is transitive if and only if its matrix $M_R = [m_{ij}]$ has the property if $m_{ij} = 1$, then $m_{jk} = 1$.

The left-hand side of this statement simply means that $(M_R)^2_{\odot}$ has a 1 in position i, k. Thus the transitivity of $\Re$ means that if $(M_R)^2_{\odot}$ has a 1 in any position, then M_R must have a 1 in the same portion. Thus, in particular, if $(M_R)^2_{\odot} = M_R$, then $\Re$ is transitive. The converse is not true.

4.13 Equivalence Relations

A relation $\Re$ on a set A is called an equivalence relation if it is **reflexive**, **symmetric** and **transitive**.

Equivalence Relations and Partitions: If P is a partition of a set A, then P can be used to construct an equivalence relation on A.

If P is a partition of A and $\Re$ be the equivalent relation determined by P, the the blocks of P can easily be described in terms of $\Re$.

If A_1 is a block of P and $a \in A_1$. We can see by definition that A_1 consists of all elements x of A with ${}_a\Re_x$. *i.e.* $A_1 = \Re(a)$.

Thus the partition P is $\{\Re(a): a \in A\}$. In other words, P consists of all distinct $\Re$-relative sets that arise from elements of A.

Partial Ordering: A relation $\Re$ is a set A is a partial ordering if it is **reflexive**, **antisymmetric** and **transitive**. Once a partial ordering is defined, the natural question arises, what is **total ordering.** A partial ordering $\Re$ is a set is a total ordering if for every pair (a, b) is $A \times A$, either ${}_a\Re_b$ or ${}_b\Re_a$

Equivalnce Class: Let $\Re$ be an equivalence relation on A and $a \in A$. Then $[a]_R = \{b \in A: {}_a\Re_b\}$ and $[a]_R$ is called R – equivalence class or simply an equivalence class generated by the element a.

Note: $[a]_R$ is written as $[a]$ itself when there is no ambiguity regarding the relation.

SOLVED EXAMPLES

Example 1

Let A be the set of all triangles in the plane and let $\Re$ be the relation on A defined as follows $\Re = \{(a, b) \in A \times A : a$ is congruent to b and let $A = Z$, the set of integers, and let R be defined by ${}_a\Re_b$ iff $a \le b$. Is $\Re$ a equivalence relation}?

Solution

It is very easy to see that $\Re$ is an equivalence relation.

Since $a \le a$, $\Re$ is reflexive. If $a \le b$, it need not follow that $b \le a$, so $\Re$ is not symmetric. Identically, $\Re$ is transitive since $a \le b$ and $b \le c \Rightarrow a \le c$ we can say that $\Re$ is not an equivalence relation. ■

Example 2

Show that congruence mod 2 is an equivalence relation.

Solution

It is clear that $a \equiv a$ *(mod 2)*. Thus $\Re$ is reflexive If $a \equiv b$ *(mod 2)*, then $a \equiv r$ *(mod 2)* and $b \equiv r$ *(mod 2)* so $b \equiv a$ *(mod 2)*. $\Re$ is symmetric.

Finally, suppose $a \equiv a$ *(mod 2)* and $b = c$ *(mod 2)* Then $a \equiv r$ *(mod 2)*, $b = r$ *(mod 2)* and $c \equiv r$ *(mod 2)*. That is, all three yield the same remainder when divided by 2. Thus, $a \equiv c$ *(mod 2)*.

Hence congruence *mode 2* is an equivalence relation. ■

4.14 Closure of Relations

When *a* relation $\mathcal{R}$ is *a* set *A* does not possess some property $\mathcal{P}$, we can add some more ordered pairs from $A \times A$ to $\mathcal{R}$, so that the extended set has that property. Among several extended sets having that property $\mathcal{P}$, we choose the smallest extended set and call it the *closure* of $\mathcal{R}$ with respect to property. *R*

Let $\mathcal{R}$ be *a* relation is a set *A* and $\mathcal{P}$ *a* property. Then *a* relation $\mathcal{R}^*$ is the closure of $\mathcal{R}$ with respect to $\mathcal{P}$ if.

- $\mathcal{R}^*$ has the property $\mathcal{P}$
- $\mathcal{R} \subseteq \mathcal{R}^*$
- If $S \subseteq A \times A$ contains $\mathcal{R}$ and has property $\mathcal{P}$, then $\mathcal{R}^* \subseteq S$. If $\mathcal{P}$ denotes the reflexive property, then $\mathcal{R}^*$ is called the *reflexive closere of* $\mathcal{R}$. Similarly we can define symmetric and transitive closures of $\mathcal{R}$.

Let *A* be *a* finite set, then

$$\Delta_A = \{(a, a): a \in A\}$$

We simply write Δ when there is so ambiguity regarding the set *A*. Δ_A is called the *diagonal relation* on *A*. This represents the equality relation.

Theorem 6: Let $\mathcal{P}$ be a partition of a set *A*. Recall that the sets in $\mathcal{P}$ are called the blocks of $\mathcal{P}$. Define the relation $\mathcal{R}$ on *A* as follows:

${}_a\mathcal{R}_b$ if and only if *a* and *b* are members of the same block. Then $\mathcal{R}$ is an equivalence relation on *A*.

Proof:

1. If $a \in A$, then clearly *a* is in the same block as itself; so ${}_a\mathcal{R}_a$.
2. If ${}_a\mathcal{R}_b$, then *a* and *b* are in the same block; so ${}_b\mathcal{R}_a$.
3. If ${}_a\mathcal{R}_b$ and ${}_b\mathcal{R}_c$, then *a*, *b*, and *c* must all lie in the same block of $\mathcal{P}$. Thus ${}_a\mathcal{R}_c$.

Since $\mathcal{R}$ is reflexive, symmetric, and transitive, $\mathcal{R}$ is an equivalence relation. $\mathcal{R}$ will be called the equivalence relation determined by $\mathcal{P}$.

Lemma: Let $\mathcal{R}$ be an equivalence relation on a set *A*, and let $a \in A$ and $b \in A$. Then ${}_a\mathcal{R}_b$ if and only is $\mathcal{R}(a) = \mathcal{R}(b)$.

Proof: First suppose that $\mathcal{R}(a) = \mathcal{R}(b)$. Since $\mathcal{R}$ is reflexive, $b \in \mathcal{R}(b)$; therefore, $b \in \mathcal{R}(a)$, so ${}_a\mathcal{R}_b$.

Conversely, suppose that ${}_a\mathcal{R}_b$. Then note that

1. $b \in \mathcal{R}(a)$ by definition; therefore, since $\mathcal{R}$ is symmetric,
2. $a \in \mathcal{R}(b)$.

We must show that $\mathcal{R}(a) = \mathcal{R}(b)$. First, choose a element $x \in \mathcal{R}(b)$. Since $\mathcal{R}$ is transitive, the fact that $x \in \mathcal{R}(b)$, together with (1) above, implies that $x \in \mathcal{R}(a)$. Thus $\mathcal{R}(b) \subseteq \mathcal{R}(a)$. Now choose $y \in \mathcal{R}(a)$. This fact and (2) above imply, as before, that $y \in \mathcal{R}(b)$. Thus $\mathcal{R}(a) \subseteq \mathcal{R}(b)$, so we must have $\mathcal{R}(a) = \mathcal{R}(b)$.

Theorem 7: Let $\mathcal{R}$ be an equivalence relation on A, and let $\mathcal{P}$ be the collection of all distinct relative sets $\mathcal{R}(a)$ for a in A. Then $\mathcal{P}$ is a partition of A, and $\mathcal{R}$ is the equivalence relation determined by $\mathcal{P}$.

Proof: According to the definition of a partition, we must show the following two properties:

(a) Every element of A belongs to some relative set.

(b) If $\mathcal{R}(a)$ and $\mathcal{R}(b)$ are not identical, then $\mathcal{R}(a) \cap \mathcal{R}(b) = \varnothing$. Now property ($a$) is true, since $a \in \mathcal{R}(a)$ by reflexivity of $\mathcal{R}$. To show property (b) we prove the following equivalent statement: If $\mathcal{R}(a) \cap \mathcal{R}(b) \neq \varnothing$, then $\mathcal{R}(a) = \mathcal{R}(b)$.

To prove this, we assume that $c \in \mathcal{R}(a) \cap \mathcal{R}(b)$. Then ${}_a\mathcal{R}_c$ and ${}_b\mathcal{R}_c$. Since $\mathcal{R}$ is symmetric, we have ${}_c\mathcal{R}_b$. Then ${}_a\mathcal{R}_c$ and ${}_c\mathcal{R}_b$, so by transitivity of $\mathcal{R}$, ${}_a\mathcal{R}_b$. Above Lemma then tells us that $\mathcal{R}(a) = \mathcal{R}(b)$. We have now proved that $\mathcal{P}$ is a partition. By Lemma above we see that ${}_a\mathcal{R}_b$ if and only if a and b belong to the same block of $\mathcal{P}$. Thus $\mathcal{P}$ determines $\mathcal{R}$, and the theorem is proved.

4.15 Manipulation and Composition of Relations

Let $\mathcal{R}$ and $\mathcal{S}$ be relations from a set A to a set B. Then, if we remember that $\mathcal{R}$ and $\mathcal{S}$ are simply subsets of $A \times B$, we can use set operations on $\mathcal{R}$ and $\mathcal{S}$. For example, the complement of $\mathcal{R}$, $\overline{\mathcal{R}}$ is referred to as the *Complementary* relation. It is, of course, a relation from A to B that can be expressed simply in terms of $\mathcal{R}$:

${}_a\overline{R}_b$ if and only if ${}_a\mathcal{R}_b$.

We can also form the intersection $\mathcal{R} \cap \mathcal{S}$ and the union $\mathcal{R} \cup \mathcal{S}$ of the relations $\mathcal{R}$ and $\mathcal{S}$. In relational terms, we see that a $(\mathcal{R} \cap \mathcal{S})$ b means that ${}_a\mathcal{R}_b$ and ${}_a\mathcal{S}_b$; a $(\mathcal{R} \cup \mathcal{S})$ b means that ${}_a\mathcal{R}_b$ or ${}_a\mathcal{S}_b$. All our set-theoretic operations can be used in this way to produce new relations. The reader should try to give a relational description of the relation $\mathcal{R} \oplus \mathcal{S}$.

A different type of operation on a relation $\mathcal{R}$ from A to B is the formation of the inverse, usually written $\mathcal{R}^{-1}$. The relation $\mathcal{R}^{-1}$ is a relation from B to A (reverse order from $\mathcal{R}$) defined by ${}_b\mathcal{R}^{-1}{}_a$ if and only if ${}_a\mathcal{R}_b$.

It is clear from this that $(\mathcal{R}^{-1})^{-1} = \mathcal{R}$. It is not hard to see that Dom $(\mathcal{R}^{-1})$ = Ran $(\mathcal{R})$ and Ran $(\mathcal{R}^{-1})$ = Dom $(\mathcal{R})$. We leave these simple facts as exercises.

Let A, B, and C are sets. $\mathcal{R}$ is a relation from A to B, and $\mathcal{S}$ is a relation from B to C. We can then define a new relation, the *composition* of $\mathcal{R}$ and $\mathcal{S}$, written $\mathcal{S} \circ \mathcal{R}$. The relation $\mathcal{S} \circ \mathcal{R}$ is a relation from A to C and is defined as follows. If a is in A and c is in C, then a $(\mathcal{S} \circ \mathcal{R})$ c if and only if for some b in B, we have ${}_a\mathcal{R}_b$ and ${}_b\mathcal{S}_c$. In other words, a is relation to c by $\mathcal{S} \circ \mathcal{R}$ if we can get from a to c in two stages: first to an intermediate vertex b by relation $\mathcal{R}$ and then from b to c by relation $\mathcal{S}$. The relation $\mathcal{S} \circ \mathcal{R}$ might be thought of as "$\mathcal{S}$ following $\mathcal{R}$"since it represents the combined effect of two relations, first $\mathcal{R}$, then $\mathcal{S}$.

For example: Let $A = \{1, 2, 3, 4\}$ $\mathcal{R} = \{(1, 2), (1, 1), (1, 3), (2, 4), (3, 2)\}$, and $\mathcal{S} = \{(1, 4), (1, 3), (2, 3), (3, 1), (4, 1)\}$. Since $(1, 2) \in \mathcal{R}$ and $(2, 3) \in \mathcal{S}$, we must have $(1, 3) \in \mathcal{S} \circ \mathcal{R}$. Similarly, since $(1, 1) \in \mathcal{R}$ and $(1, 4) \in \mathcal{S}$, we see that $(1, 4) \in \mathcal{S} \circ \mathcal{R}$. Proceeding in this way, we find that $\mathcal{S} \circ \mathcal{R} = \{(1, 4), (1, 3), (1, 1), (2, 1), (3, 3)\}$.

SOLVED EXAMPLES

Example 1

If $A = \{a, b, c, d, e\}$ and $\mathcal{R}$ and $\mathcal{S}$ two relations on A. Find $\overline{\mathcal{R}}$ and $\mathcal{R}^{-1}$. Also prove that $\mathcal{R}$ is symmetric iff $\mathcal{R} = \mathcal{R}^{-1}$.

Solution

Let $A = \{a, b, c, d, e\}$ and let $\mathcal{R}$ and $\mathcal{S}$ be two relations on A whose corresponding digraphs are shown in Fig. 4. 8. Then the reader can verify the following facts:

$$\overline{\mathcal{R}} = \{(a, a), (b, b), (a, c), (b, a), (c, b), (c, d), (c, e), (c, a), (d, b), (d, a), (d, e), (e, b), (e, a), (e, d), (e, c)\}$$

$$\mathcal{R}^{-1} = \{(b, a), (e, b), (c, c), (c, d), (d, d), (d, b), (c, b), (d, a), (e, e), (e, a)\}$$

$$\mathcal{R} \cap \mathcal{S} = \{(a, b), (b, e), (c, c)\}.$$

The above example we Illustrates some general facts. Recalling the operations on Boolean matrices, we can show that if $\mathcal{R}$ and $\mathcal{S}$ are relations on set A, then

$$M_R \cap_s = M_R \wedge M_S$$

$$M_R \cup_s = M_R \vee M_S$$
$$M_{R^{-1}} = (M_R)^T.$$

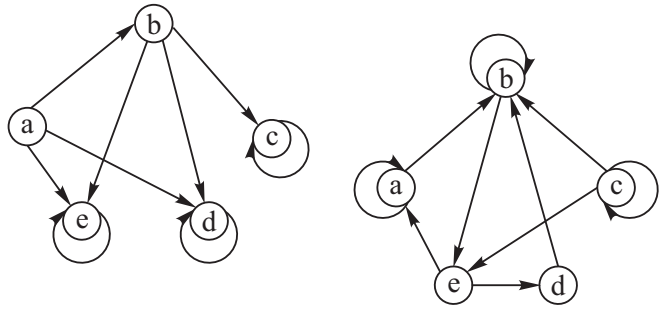


Fig. 4.8

Moreover, if M is a Boolean matrix, define the complement $\overline{M}$ of M as the matrix obtained from M by replacing every 1 in M by a 0 and every 0 by a 1. Thus, if

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

then
$$\overline{M} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

We can also show that if $\Re$ is a relation on a set A. then

$$M_{\overline{R}} = \overline{M}_R$$

We know that a symmetric relation is a relation $\Re$ such that $M_R = (M_R)^T$, and since $(M_R)^T = M_{R^{-1}}$, we see that $\Re$ is symmetric if and only if $\Re = \Re^{-1}$.

■

Example 2

Explain (i) Reflexive closure and (ii) symmetric cloure by taking suitable examples.

Solution

(*i*) Suppose that $\Re$ is a relation on a set A and $\Re$ is not reflexive. This can only occur because some pairs of the diagonal relation Δ are not in $\Re$. Thus $\Re_1 = \Re \cup \Delta$ is the smallest reflexive relation on A containing $\Re$, that is, the *reflexive closure* of $\Re$ is $\Re \cup \Delta$.

(*ii*) Suppose now that $\mathcal{R}$ is a relation on A that is not symmetric. Then there must exist pairs (x, y) in $\mathcal{R}$ such that (y, x) is not in $\mathcal{R}$. Of course $(y, x) \in \mathcal{R}^{-1}$, so if $\mathcal{R}$ is to be symmetric we must add all pairs from $\mathcal{R}^{-1}$; that is, we must enlarge $\mathcal{R}$ to $\mathcal{R} \cup \mathcal{R}^{-1}$. Clearly, $(\mathcal{R} \cup \mathcal{R}^{-1})^{-1} = \mathcal{R} \cup \mathcal{R}^{-1}$, so $\mathcal{R} \cup \mathcal{R}^{-1}$ is the smallest symmetric relation containing $\mathcal{R}$; that is, $\mathcal{R} \cup \mathcal{R}^{-1}$ is the *symmetric closure* of $\mathcal{R}$.

If $A = \{a , b\ c , d\}$ and $\mathcal{R} = \{(a, b) , (b, c), (a, c), (c, d\}$ then

$\mathcal{R}^{-1} = \{(b, a), (c, b), (c, a), (d, c)\}$, so the symmetric closure of $\mathcal{R}$ is

$\mathcal{R} \cup \mathcal{R}^{-1} = \{(a, b), (b, a), (b, c), (c, b), (a, c), (c, a), (c, d), (d, c)\}$.

■

Example 3

Let $A = \{a, b, c\}$ and let $\mathcal{R}$ and $\mathcal{S}$ be relations A whose matrices are

$$M_R = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \quad M_s = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Solution

We see from the matrices that

$(a, a) \in \mathcal{R}$ and $(a, a) \in \mathcal{S}$, so $(a, a) \in \mathcal{S} \circ \mathcal{R}$

$(a, c) \in \mathcal{R}$ and $(c, a) \in \mathcal{S}$, so $(a, a) \in \mathcal{S} \circ \mathcal{R}$

$(a, c) \in \mathcal{R}$ and $(c, c) \in \mathcal{S}$, so $(a, c) \in \mathcal{S} \circ \mathcal{R}$.

It is easily seen that $(a, a) \notin \mathcal{S} \circ \mathcal{R}$ since, if we had $(a, x) \in \mathcal{R}$ and $(x, b) \in \mathcal{S}$, matrix M_R tells us that x would have to be a or c; but matrix $M_{\mathcal{S}}$ tells us that neither (a, b) nor (c, b) is an element of $\mathcal{S}$.

We see that first row of $M_{S \circ R}$ is 1 0 1. The reader may show by similar analysis that

$$M_{\mathcal{R}} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

We note that $M_{\mathcal{S} \circ \mathcal{R}} = M_R \odot M_{\mathcal{S}}$ (verify).

Note: The above example illustrates a general and useful fact.

Let A, B, and C be finite sets with n, p, and m elements, respectively, let $\mathcal{R}$ be a relation from A to B, and let $\mathcal{S}$ be a relation from B to C. Then $\mathcal{R}$ and $\mathcal{S}$ be a relation from B to C. Then $\mathcal{R}$ and $\mathcal{S}$ have Boolean matrices M_R and $M_{\mathcal{S}}$ with respective sizes $n \times p$ and $p \times m$. Then $M_R \odot M_{\mathcal{S}}$ can be computed, and it equals $M_{\mathcal{S} \circ \mathcal{R}}$.

To see this, let $A = \{a_1,..., a_n\}$, $B = \{b_1,..., b_p\}$, and $C = \{c_1,..., c_m\}$. Also suppose that $M_R = [r_{ij}]$, $M_S = [s_{ij}]$, and $M_{S \circ R} = [t_{ij}]$. Then $t_{ij} = 1$ if and only if $(a_i, c_j) \in S \circ \mathcal{R}$, which means that for some k, $(a_i, b_k) \in R$ and $(b_k, c_j) \in S$. In other words, $r_{ik} = 1$ and $s_{kj} = 1$ for some k between 1 and p. This condition is identical to the condition needed for $M_R \odot M_{\mathcal{S}}$ are equal.

In the special case where $\mathcal{R}$ and $\mathcal{S}$ are equal, we have $\mathcal{S} \circ \mathcal{R} = \mathcal{R}^2$ and $M_{\mathcal{S} \circ \mathcal{R}} = M_R \odot M_R$. ■

Example 4

Let $A = \{1, 2, 3, 4\}$ and let $\mathcal{R} = \{(1, 2), (2, 3), (3, 4), (2, 1)\}$ Find the transitive closure of $\mathcal{R}$.

Solution Method I

The digraph of $\mathcal{R}$ as shown in Fig. 4.9 illustrates the problem. Since $\mathcal{R}^\infty$ is transitive closure, we can proceed it geometrically by computing all paths. We can see from vertix 1 we have paths to vertices 2, 3, 4 and 1. The path from 1 to 1 proceeds from 1 to 2 to 1. This shows that the ordered pairs (1, 1), (1, 2), (1, 3), and (1, 4) are in $\mathcal{R}^\infty$. Starting from vertex 2, we have paths to vertices 2, 1, 3, and 4; so the ordered pairs (2, 1), (2, 2), (2, 3), and (2, 4) are in $\mathcal{R}^\infty$. The only other path is from vertex 3 to vertex 4, so we have

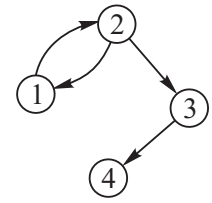


Fig. 4.9

$\mathcal{R}^\infty = \{(1, 1), (1, 2), (1, 3), (1, 4), (2, 1), (2, 2), (2, 3), (2, 4), (3, 4)\}$.

Method II. The matrix of $\mathcal{R}$ is

$$M_{\mathcal{R}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

We may proceed algebraically and compute the powers of M_R. Thus

$$(M_{\mathcal{R}})^2_{\odot} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad (M_{\mathcal{R}})^3_{\odot} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$(M_{\mathcal{R}})^4_{\odot} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Continuing in this way, we can see that $(M_{\mathcal{R}})^n_{\odot}$ equals $(M_{\mathcal{R}})^2_{\odot}$ if n is even and equals $(M_{\mathcal{R}})^3_{\odot}$ if n is odd and greater than 1. Thus

$$M_{\mathcal{R}\infty} = M_{\mathcal{R}} \vee (M_{\mathcal{R}})^2_{\odot} \vee (M_{\mathcal{R}})^3_{\odot} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

and this gives the same relation as Method I. ■

Theorem 8: Suppose that $\mathcal{R}$ and $\mathcal{S}$ are relations from A to B.

(a) If $\mathcal{R} \subseteq \mathcal{S}$, then $\mathcal{R}^{-1} \subseteq \mathcal{S}^{-1}$.

(b) If $\mathcal{R} \subseteq \mathcal{S}$, then $\overline{\mathcal{S}} \subseteq \overline{\mathcal{R}}$

(c) $(\mathcal{R} \cap \mathcal{S})^{-1} = \mathcal{R}^{-1} \cap \mathcal{S}^{-1}$ and $(\mathcal{R} \cup \mathcal{S})^{-1} = \mathcal{R}^{-1} \cup \mathcal{S}^{-1}$.

(d) $(\overline{\mathcal{R} \cap \mathcal{S}}) = \overline{\mathcal{R}} \cup \mathcal{S}$ and $(\overline{\mathcal{R} \cup \mathcal{S}}) = \overline{\mathcal{R}} \cap \overline{\mathcal{S}}$.

Proof: Parts (b) and (d) are special cases of general set properties already proved.

We now prove part (a). Suppose that $\mathcal{R} \subseteq \mathcal{S}$ and let $(a, b) \in \mathcal{R}^{-1}$. Then $(b, a) \in \mathcal{R}$, so $(b, a) \in \mathcal{S}$. This, in turn implies that $(a, b) \in \mathcal{S}^{-1}$. Since each element of $\mathcal{R}^{-1}$ is in $\mathcal{S}^{-1}$, we are done.

We next prove part (c). For the first part, suppose that $(a, b) \in (\mathcal{R} \cap \mathcal{S})^{-1}$. Then $(b, a) \in \mathcal{R} \cap \mathcal{S}$, so $(b, a) \in \mathcal{R}$ and $(b, a) \in \mathcal{S}$. This means that $(a, b) \in \mathcal{R}^{-1}$ and $(a, b) \in \mathcal{S}^{-1}$, so $(a, b) \in \mathcal{R}^{-1} \cap \mathcal{S}^{-1}$. Then converse containment can be proved by reversing the steps. *A* similar argument works to show that $(\mathcal{R} \cup \mathcal{S})^{-1} = \mathcal{R}^{-1} \cup \mathcal{S}^{-1}$.

Theorem 9: Let $\mathcal{R}$ and $\mathcal{S}$ be relations on a set A.

(a) If $\mathcal{R}$ is reflexive, so is $\mathcal{R}^{-1}$.

(b) If $\mathcal{R}$ and $\mathcal{S}$ are reflexives, then so are $\mathcal{R} \cap \mathcal{S}$ and $\mathcal{R} \cup \mathcal{S}$.

(*c*) $\mathcal{R}$ is reflexive if and only if $\overline{\mathcal{R}}$ is irreflexive.

Proof: Let Δ be the equality relation on. *A*. We know that $\mathcal{R}$ is reflexive if and only if $\Delta \subseteq \mathcal{R}$. *C*learly, $\Delta = \Delta^{-1}$, so if $\Delta \subseteq \mathcal{R}$, then $\Delta = \Delta^{-1} \subseteq \mathcal{R}^{-1}$ by Theorem, so $\mathcal{R}^{-1}$ is also reflexive. This proves part (*a*). To prove part (*b*), we note that if $\Delta \subseteq \mathcal{R}$ and $\Delta \subseteq \mathcal{S}$, then $\Delta \subseteq \mathcal{R} \cap \mathcal{S}$ and $\Delta \subseteq \mathcal{R} \cap \mathcal{S}$. To show part (*c*), we note that a relation $\mathcal{S}$ is irreflexive if and only if $\mathcal{S} \cap \Delta = \varnothing$. Then $\mathcal{R}$ is reflexive if and only if $\Delta \subseteq \mathcal{R}$ if and only if $\Delta \cap \mathcal{R} = \varnothing$ if and only if $\overline{\mathcal{R}}$ is irreflexive.

Theorem 10: Let $\mathcal{R}$ and $\mathcal{S}$ be relations on A.

If $\mathcal{R}$ is symmetric, so are $\mathcal{R}^{-1}$ and $\overline{\mathcal{R}}$.

Proof: If $\mathcal{R}$ is symmetric, $\mathcal{R} = \mathcal{R}^{-1}$ and thus $(\mathcal{R}^{-1})^{-1} = \mathcal{R} = \mathcal{R}^{-1}$, which means that $\mathcal{R}^{-1}$ is also symmetric. Also, $(a, b) \in (\overline{\mathcal{R}})^{-1}$ if and only if $(b, a) \in \overline{\mathcal{R}}$ if and only if $(b, a) \notin \mathcal{R}$ if and only if $(a, b) \notin \mathcal{R}^{-1} = \mathcal{R}$ if and only if $(a, b) \in \overline{\mathcal{R}}$, so $\overline{\mathcal{R}}$, is symmetric

Theorem 11: Let $\mathcal{R}$ and $\mathcal{S}$ be relations on A.

(*a*) $(\mathcal{R} \cap \mathcal{S})^2 \subseteq \mathcal{R}^2 \cap \mathcal{S}^2$.

(*b*) If $\mathcal{R}$ and $\mathcal{S}$ are transitive, so is $\mathcal{R} \cap \mathcal{S}$.

(*c*) If $\mathcal{R}$ and $\mathcal{S}$ are equivalence relations, so is $\mathcal{R} \cap \mathcal{S}$.

Proof: We prove part (*a*) geometrically. We have ${}_a(\mathcal{R} \cap \mathcal{S})_b$ if and only if there is a path of length 2 from *a* to *b* in $\mathcal{R} \cap \mathcal{S}$. Both edges of this path lie in $\mathcal{R}$ and in $\mathcal{S}$, so ${}_a\mathcal{R}^2{}_b$ and ${}_a\mathcal{S}^2{}_b$, which implies that ${}_a(\mathcal{R}^2 \cap \mathcal{S}^2)_b$. To show part (*b*), recall that a relation *T* is transitive if and only if $T^2 \subseteq T$. If $\mathcal{R}$ and $\mathcal{S}$ are transitive, then $\mathcal{R}^2 \subseteq \mathcal{R}$, $\mathcal{S}^2$, $\subseteq \mathcal{S}$ so $(\mathcal{R} \cap \mathcal{S})^2 \subseteq \mathcal{R}^2 \cap \mathcal{S}^2$ [by part (*a*)] $\subseteq \mathcal{R} \cap \mathcal{S}$, so $\mathcal{R} \cap \mathcal{S}$ is transitive. We next prove part (*c*). Relations $\mathcal{R}$ and $\mathcal{S}$ are each reflexive, symmetric, and transitive. The same properties hold for $\mathcal{R} \cap \mathcal{S}$. Hence $\mathcal{R} \cap \mathcal{S}$ is an equivalence relation.

Theorem 12: Let $\mathcal{R}$ be a relation from *A* to *B* and let $\mathcal{S}$ be a relation from *B* to *C*. Then, if A_1 is any subset of *A*, we have

$$(\mathcal{S} \circ \mathcal{R})(A_1) = \mathcal{S}(\mathcal{R}(A_1)).$$

Proof: If an element $z \in C$ is in $(\mathcal{S} \circ \mathcal{R})(A_1)$, then $x\ (\mathcal{S} \circ \mathcal{R})\ z$ for some x in A_1. By the definition of composition, this means that ${}_x\mathcal{R}_y$ and ${}_y\mathcal{S}_z$ some y in *B*. Thus $y \in \mathcal{R}(x)$, so $z \in \mathcal{S}(\mathcal{R}(x))$. Since $\{x\} \subseteq A_1$. $\mathcal{S}(\mathcal{R}(x)) \subseteq \mathcal{S}(\mathcal{R}(A_1))$. Hence $z \in \mathcal{S}(\mathcal{R}(A_1))$, so $(\mathcal{S} \circ \mathcal{R})(A_1) \subseteq \mathcal{S}(\mathcal{R}(\mathrm{A}_1))$.

Conversely, suppose that $z \in \mathcal{S}(\mathcal{R}(A_1))$. Then $z \in \mathcal{S}(y)$ for some y in $\mathcal{R}(A_1)$ and similarly, $y \in \mathcal{R}(x)$ for some x in A_1. This means that ${}_x\mathcal{R}_y$ and

${}_y\mathbb{S}_z$, so $x\ (\mathbb{S}\ o\ \mathbb{R})z$. Thus $z \in (\mathbb{S}\ o\ \mathbb{R})\ (A_1)$, so $\mathbb{S}(\mathbb{R}(A_1)) \subseteq (\mathbb{S}\ o\ \mathbb{R})\ (A_1)$. This proves the theorem.

Theorem 13: Let A, B, C, and D be sets: $\mathbb{R}$ a relation from A to B $\mathbb{S}$ a relation from B to C, and $\mathbb{T}$ a relation from C to D. Then $\mathbb{T}o\ (\mathbb{S}\ o\ \mathbb{R}) = (\mathbb{T}o\ \mathbb{S})\ o\ \mathbb{R}$.

Proof: The relations $\mathbb{R}$, $\mathbb{S}$, and $\mathbb{T}$ are determined by their Boolean matrices $M_{\mathbb{R}}$, $M_{\mathbb{S}}$, and $M_{\mathbb{T}}$, respectively the matrix of the composition is the Boolean matrix product; that is, $M_{\mathbb{S}\ o\ \mathbb{R}} = M_{\mathbb{R}} \odot M_{\mathbb{S}}$. Thus

$$M_{\mathbb{T}\ (\mathbb{S}\ o\ \mathbb{R})} = M_{\mathbb{S}\ o\ \mathbb{R}} \odot M_{\mathbb{T}} = (M_{\mathbb{R}} \odot M_{\mathbb{S}}) \odot M_{\mathbb{T}}$$

Similarly,

$$M\ (_{\mathbb{T}\ \mathbb{S}\ o\ \mathbb{R}}) = M_{\mathbb{R}} \odot (M_{\mathbb{S}} \odot M_{\mathbb{T}}).$$

Since Boolean matrix multiplication is associative, we must have

$$(M_{\mathbb{R}} \odot M_{\mathbb{S}}) \odot M = M_{\mathbb{R}} \odot (M_{\mathbb{S}} \odot M_{\mathbb{T}}).$$

and therefore

$$M_{\mathbb{T}\ o(\mathbb{S}\ o\ \mathbb{R})} = M(\mathbb{T}\ o\mathbb{S})\ o\ \mathbb{R}$$

Then $\mathbb{T}\ o\ (\mathbb{S}\ o\ \mathbb{R}) = (\mathbb{T}\ o\mathbb{S})\ o\ \mathbb{R}$

Since these relations have the same matrices. In general, $\mathbb{R}\ o\ \mathbb{S} \neq \mathbb{S}\ o\ \mathbb{R}$,

Theorem 14: Let $\mathbb{R}$ be a relation on a set A. Then $\mathbb{R}^\infty$ is the transitive closure of $\mathbb{R}$.

Proof: We recall that if a and b are in the set A, then ${}_a\mathbb{R}_b^\infty$ if and only if there is a path in $\mathbb{R}$ from a to b. Now $\mathbb{R}^\infty$ is certainly transitive since, if ${}_a\mathbb{R}_b^\infty$ and ${}_a\mathbb{R}_c^\infty$, the composition of the paths from a to b and from b to c form a path from a to c in $\mathbb{R}$, and so ${}_a\mathbb{R}_c^\infty$. To show that $\mathbb{R}^\infty$ is the smallest transitive relation containing $\mathbb{R}$, we must show that if $\mathbb{S}$ is any transitive relation on A and $\mathbb{R} \subseteq \mathbb{S}$, then $\mathbb{R}^\infty \subseteq \mathbb{S}$. if $\mathbb{S}$ is transitive, then $\mathbb{S}^n \subseteq \mathbb{S}$ for all n; that is, if a and b are connected by a path of length n, then ${}_a\mathbb{S}_b$. It follows that $\mathbb{S}^\infty = \cup_{n=1}^\infty\ \mathbb{S}^n \subseteq \mathbb{S}$. It is also true that if $\mathbb{R} \subseteq \mathbb{S}$, then $\mathbb{R}^\infty \subseteq \mathbb{S}^\infty$, since any path in $\mathbb{R}$ is also a path in $\mathbb{S}$. Putting these facts together, we see that if $\mathbb{R} \subseteq \mathbb{S}$ and $\mathbb{S}$ is transitive on A, then $\mathbb{R}^\infty \subseteq \mathbb{S}^\infty \subseteq \mathbb{S}^\infty$. This means that $\mathbb{R}^\infty$ is the smallest of all transitive relations on A that contain $\mathbb{R}$.

Theorem 15: Let A be a set with $|\ A\ | = n$, and let R be a relation on A.

Then

$$\mathbb{R}^\infty = \mathbb{R} \cup \mathbb{R}^2 \cup \ldots \cup \mathbb{R}^n.$$

In other words, powers of $\mathcal{R}$ greater than n are not needed to compute $\mathcal{R}^{\infty}$.

Proof: Let a and b be in A, and suppose that $a, x_1, x_2, ..., x_m, b$ is a path from a to b in $\mathcal{R}$; that is, $(a, x_1), (x_1, x_2), ...(x_m, b)$ are all in $\mathcal{R}$. If x_i and x_j are equal, say $i < j$, then the path can be divided into three sections. First, a path from a to xi, then a path from x_i to x_j, and finally a path from x_j to b. The middle path is a cycle, since $x_i = x_j$, so we simply leave it out and put the first two paths together. This gives us a shorter path from a to b (see) Fig. 4.10.

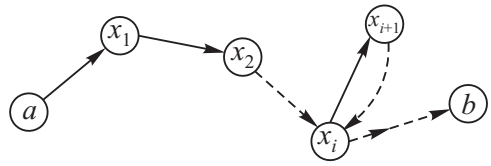


Fig. 4.10

Now let $a, x_1, x_2, ...,x_k, b$ be the shortest path from a to b. If $a \neq b$, then all vertices $a, x_1, x_2, ..., x_k, b$ are distinct. Otherwise, the preceding discussion shows that we could find a shorter path. Thus the length of the path is at most $n - 1$ (since $| A | = n$). If $a = b$, then for similar reasons, the vertices $a, x_1, x_2, ... x_k$ are distinct, so the length of the path is at most n. In other words, if a $\mathcal{R}^{\infty}$ b, then a $\mathcal{R}^k$ b, for some k, $1 \leq k \leq n$. Thus $\mathcal{R}^{\infty} = \mathcal{R} \cup \mathcal{R}^2 \cup ... \cup \mathcal{R}^n$.

4.16 Warshall's Algorithm

Since $\mathcal{R}^{\infty} = \mathcal{R}^*$, the connectivity relation algorithem can be used to compute $M_{\mathcal{R}^*}$, but it is not efficient, especially when $M_{\mathcal{R}^*}$ is very large. *A* better method to find $\mathcal{R}^*$ is **Warshall's algorithm**, named is the honour of **Stephen Warshall,** who discovered it in 1962.

Let $\mathcal{R}$ be *a* relation on *a* set $A = \{a_1, a_2, ..., a_m\}$. If $x_1, x_2, ..., x_m$ is a path $\mathcal{R}$, then any vertices other then x, and x_m are called *interior vertices* of the path.

Now, for $1 \subseteq k \leq n$, we define *a* Boolean matrix W_k as follows. W_k has *a* 1 is position i, j if and only if there is *a* path from *ai* to *aj* is $\mathcal{R}$ whose interior vertices, if any, come from the set $\{a_1, a_2, ..., a_k\}$.

Since any vertex must come from the set $\{a_1, a_2, ..., a_n\}$, it follows that the matrix W_n has a_1 in position i, j if and only if some path in $\mathcal{R}$

connects a_i with a_j. In other words, $W_n = M_{R\infty}$. If we define W_0 to be $M_{\mathbb{R}}$, then we will have sequence W_0, W_1,...,W_n whose first term is $M_{\mathbb{R}}$ and whose last term is $M_{\mathbb{R}\infty}$. We will show how to compute each matrix W_k from the previous matrix W_{k-1}. Then we can begin with matrix of $\mathbb{R}$ and proceed one step at a time until, in n steps, we reach the matrix of $\mathbb{R}^\infty$. This procedure is called **Warshall's algorithm.** The matrices W_k are different from power of the matrix $M_{\mathbb{R}}$, and this difference results in a considerable savings of steps in the computation of the transitive closure of $\mathbb{R}$.

Suppose that $W_k = [t_{ij}]$ and $W_{k-1} = [s_{ij}]$. If $t_{ij} = 1$, then there must be path from a_i to a_j whose interior vertices come from the set [a_1, a_2,..., a_k]. If the vertex a_k is not an interior vertex of this path, then all interior vertices must actually come from the set [a_1, a_2,..., a_{k-1}], so $s_{ij} = 1$. If a_k is an interior vertex of the path, then the situation is as shown in Figure 4.11. We may assume that all interior vertices are distinct. Thus a_k appears only once in the path, so all interior vertices of subpaths 1 and 2 must come from the set [a_1, a_2,..., a_{k-1}]. This means that $s_{ik} = 1$ and $s_{kj} = 1$.

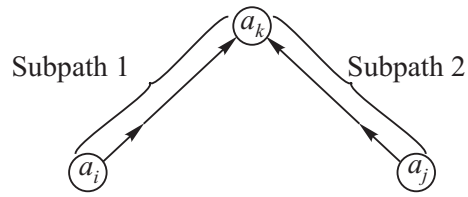


Fig. 4.11

Thus $t_{ij} = 1$ if and only if either

1. $s_{ij} = 1$ or
2. $s_{ik} = 1$ and $s_{kj} = 1$.

This is the basis for Warshall's algorithm. If W_{k-1} has a 1 in position i, j the, by (1) so will W_k, By (2), a new 1 can be added in position i, j of W_k if and only if column k of W_{k-1} has a 1 in position i and row k of W_{k-1} has a 1 in position j. Thus we have the following procedure for computing W_k from W_{k-1}.

Step 1: First transfer to W_k all 1's in W_{k-1}.

Step 2: List the locations p_1, p_2,..., in column k of W_{k-1}, where the entry is 1, and the locations q_1, q_2,..., in row k of W_{k-1}, where the entry is 1.

Step 3: Put 1's in all the positions p_i, q_j of W_k (if they are not already there).

SOLVED EXAMPLES

Example 1

Using Warshall's algorithm, find the transitive closure of the relation $R = \{(a, b), (b, a), (b, c)\}$ *on* $A = \{a, b, c\}$.

Solution

Step 1: Find W_0.

$$W_0 = M_R = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Step 2: Find W_1.

If the ijth element of W_0 is 1, the ijth element of W_1 is also 1. In other words, every 1 in W_0 stays in W_1. To find the remaining 1's in W_1, locate the 1's in column 1 (= k); there is just one 1; it occurs in position $i = 2$. Now locate the 1's in row 1 (= k). Again, their is just one 1, namely, in position $j = 2$. Therefore, the ijth entry in W_1 should be 1, where $i = 2$ and $j = 2$. Thus

$$W_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Step 3: Find W_2.

Again, all the 1's in W_1 stay in W_2. To find the other 1's, if any, locate the 1's in column 2 (= k) and row 2 (= k). They occur in positions 1 and 2 of column 2 and in positions 1, 2, and 3 of row 2, so the ijth entry for W_2 must be 1, where $i = 1, 2$ and $j = 1, 2, 3$. So change the 0's in such locations of W_1 to 1's. Thus

$$W_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

Step 4: Find W_3.

All the 1's W_2 remain in W_3. To find the remaining 1's, if any, locate the 1's in column 3 – namely, positions 1 and 2 – and the 1's in row 3. Because no 1's appear i row 3, we get no new 1's, so $W_3 = W_2$.

Since A contains three elements, $W_{R*} = W_3$. Thus,

$$W_{R*} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

■

Example 2

Using Warshall's algorithm, find the transitive closure of the relation R = {(a, a), (a, b), (a, d), (b, a), (c, b), (c, c), (d, b), (d, c), (d, d)} on {a, b, c, d}.

Solution

Step 1: Find W_0.

$$W_0 = M_R = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

Step 2: Find W_1.

Locate the 1's in column 1 and row 1; positions 1 and 2 in column 1; and positions 1, 2 and 4 in row 1. Therefore, W_1 should contain a 1 in locations (1, 1), (1, 2), (1, 4), (2, 1), (2, 2), and (2, 4):

$$W_1 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

(All the 1's in W_0 remain in W_1).

Step 3: Find W_2.

Locate the 1's in column 2 and in row 2; positions 1, 2, 3, and 4 in column 2, and position 1, 2, and 4 in row 2. So W_2 should contain a 1 in locations (1, 1), (1, 2), (1, 4), (2, 1), (2, 2), (2, 4), (3, 1), (3, 2), (3, 4), (4, 1), (4, 2), and (4, 4).

Again, since all the 1's in W_1 stay in W_2,

$$W_1 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Step 4: Find W_3.

The 1's of column 3 occur in positions 3 and 4; those of row 3 in positions 1, 2, 3, and 4. Consequently, W_3 should contain a 1 in locations (i, j) where $i = 3, 4$ and $j = 1, 2, 3, 4$:

$$W_3 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Step 5: Find W_4.

The 1's of column 4 appear in positions 1, 2, 3, and 4; the 1's of row 4 in positions 1, 2, 3, and 4. So W_4 should contain a 1 in locations (i, j) where $i = 1, 2, 3, 4$ and $j = 1, 2, 3, 4$:

$$W_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Since $M_{R^*} = W_4$, this is the adjacency matrix of the transitive closure. (Finding the connectivity relation of R will verify this). ■

Example 3

Let $A = \{(1, 2, 3, 4, 5)\}$, $\mathcal{R} = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 3), (3, 4), (4, 3), (4, 4), (5, 5)\}$, and $\mathcal{S} = (1, 1), (2, 2), (3, 3), (4, 4), (4, 5), (5, 4), (5, 5)\}$. The reader may verify that both $\mathcal{R}$ and $\mathcal{S}$ are equivalence relations. The partition $A/\mathcal{R}$ of A corresponding to $\mathcal{R}$ is $\{(1, 2), (3, 4), (5)\}$, and the partition $A/\mathcal{S}$ of A corresponding to $\mathcal{S}$ is $\{(1), (2), (3), (4, 5)\}$. Find the smallest equivalence relation containing $\mathcal{R}$ and $\mathcal{S}$, and compute the partition of A that it produces.

Solution

We have

$$M_{\mathcal{R}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } M_{\mathcal{S}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

so $$M_{\mathcal{R} \cup \mathcal{S}} = M_{\mathcal{R}} \cup M_{\mathcal{S}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Step 1: We now compute $M_{(\mathcal{R} \cup \mathcal{S})^\infty}$ by Warshall's algorithm. First, $W_0 = M_{\mathcal{R} \cup \mathcal{S}}$. We next compute W_1, so $k = 1$. Since W_0 has 1's in locations 1 and 2 of column 1 and in locations 1 and 2 of row 1, we find that no new 1's must be adjoined to W_1. Thus

$$W_1 = W_0.$$

Step 2: We now compute W_2, so $k = 2$. Since W_1 has 1's in locations 1 and 2 of column 2 and in locations 1 and 2 of row 2, we find that no new 1's must be added to W_1. Thus

$$W_2 = W_1.$$

Step 3: We next compute W_3, so $k = 3$. Since W_2 has 1's in locations 3 and 4 of column 3 and in locations 3 and 4 of row 3, we find that no new 1's must be added to W_2. Thus

$$W_3 = W_2.$$

Step 4: Things change when we now compute W_4. Since W_3 has 1's in locations 3, 4, and 5 of column 4 and in locations 3, 4, and 5 of row 4, we must add new 1's to W_3 in position 3, 5, and 5, 3, and 5, 3. Thus

$$W_4 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

The reader may verify that $W_5 = W_4$ and thus $(\mathcal{R} \cup \mathcal{S})^\infty = \{(1, 1), (1, 2), (2, 1), (2, 2), (3, 3), (3, 4), (3, 5), (4, 3), (4, 4), (4, 5), (5, 3), (5, 4), (5, 5)\}$.

The corresponding partition of A is then (verify) $\{(1, 2), (3, 4, 5)\}$.

■

Example 4

Compute the transitive closure of the relation $\{(a_1, a_1), (a_1, a_4), (a_2, a_2), (a_3\ a_4), (a_3, a_5), (a_4, a_1), (a_5, a_2), (a_5\ a_5)\}$.

Solution

$W_0 = M_{\mathrm{R}}$ is given by.

$$W_0 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Step 1: $(W_0)_{21} = (W_0)_{31} = (W_0)_{51} = 0$.

Hence the row operations for the 2nd, 3rd and 5th rows of W_1 are R_2, R_3, R_5. As $(W_0)_{11} = (W_0)_{41} = 1$

The row operations for 1st and 4th rows of W_1 are $R_1 \vee R_1 = R_1$ and $R_4 \vee R_1$ so that

$$W_1 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} R_1 \vee R_1 = R_1 \\ R_2 \\ R_3 \\ R_4 \vee R_1 \\ R_5 \end{matrix}$$

Step 2: As $(W_1)_{12} = (W_1)_{23} = (W_1)_{42} = 0$;

Row operation for 1st, 3rd and 4th rows are R_1, R_3, R_4. For the second and 5th rows, row operations are $R_2 \vee R_2$ and $R_5 \vee R_2$, so that

$$W_2 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} R_1 \\ R_2 \vee R_2 = R_2 \\ R_3 \\ R_4 \\ R_5 \vee R_2 \end{matrix}$$

Step 3: As $(W_2)_{13} = (W_2)_{23} = (W_2)_{33} = (W_2)_{43} = (W_2)_{53} = 0$ $W_3 = W_2$ as $(W_3)_{24} = (W_3)_{54} = 0$. Row operations for 2nd and 5th rows are R_2 and R_5 As $(W_3)_{14} = (W_3)_{34} = (W_3)_{44} = 1$.

Row operations for 1st, 3rd and 4th rows are $R_1 \vee R_4$, R_3 R_4 and $R_4 \vee R_4$

$$W_4 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} R_1 \vee R_4 \\ R_2 \\ R_3 \vee R_4 \\ R_4 \vee R_4 = R_4 \\ R_5 \end{matrix}$$

Step 4: As $(W_4)_{15} = (W_4)_{25} = (W_4)_{45} = 0$ and $(W_4)_{35} = (W_4)_{55} = 1$. The row operations are R_1, R_2, $R_3 \vee R_5$, R_4, $R_5 \vee R_5$

$$W_5 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Hence the transitive closure of R is given by matrix W_5.

So, $R^* = \{(a_1, a_1), (a_1, a_4), (a_2, a_2), (a_3, a_1), (a_3, a_2), (a_3, a_4), (a_3, a_5), (a_4, a_1), (a_4, a_4), (a_5, a_2), (a_5, a_5)\}$. ■

Example 5

Compute the transitive closure of the relation $R = \{(a_1, a_4), (a_1, a_5), (a_1, a_6), (a_6, a_2), (a_6, a_3), (a_6, a_5)\}$ by using Warshall's algorithm.

Solution

$W_0 = M_R$ is given by

$$W_0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Step 1: $(W_0)_{11} = (W_2)_{21} = \ldots\ldots = (W_0)_{61} = 0$. Hence all the rows are unchanged and $W_1 = W_0$

Step 2: $(W_1)_{12} = (W_1)_{22} = \ldots\ldots = \ldots\ldots (W_1)_{52} = 0$. Hence for W_2, the row operations for the first five rows are R_1, R_2,..., R_5. $(W_1)_{62} = 1$. Row operation for 6th row = $R_6 \vee R_2$

$$W_2 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \begin{matrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \vee R_2 = R_6 \end{matrix}$$

Step 3: $(W_2)_{13} = (W_2)_{23} = \ldots = (W_2)_{53} = 0$. Hence for W_3, row operations for the first five rows are R_1, R_2, R_3, R_4, R_5, as $(W_2)_{63} = 1$. Row operations for 6th row = $R_6 \vee R_3 = R_6$. Hence $W_3 = W_2$.

Step 4: As $(W_3)_{24} = (W_3)_{34} = \ldots = (W_3)_{64} = 0$. The row operations for 2nd to 6th rows are $R_2, R_3, R_4, \ldots, R_6$. As $(W_3)_{14} = 1$., row operation for the first row = $R_1 \vee R_4 = R_1$. Hence $W_4 = W_3$.

Step 5: As $(W_4)_{25} = (W_4)_{35} = (W_4)_{45} = (W_5)_{55} = 0$, row operations for 2nd to 5th rows are R_2, R_3, R_4, R_5 as $(W_4)_{15} = (W_4)_{65} = 1$., row operations for first and 6th rows are $R_1 \vee R_5$ and $R_6 \vee R_5$ *i.e.* R_1 and R_6. Hence $W_5 = W_4 = W_2$.

Step 6: As $(W_5)_{26} = (W_5)_{36} = (W_5)_{46} = (W_5)_{56} = (W_6)_{66} = 0$ row operations for 2nd to 6th rows are $R_2, R_3 \ldots, R_6$. As $(W_5)_{16} = 1$., the row operation for the first row = $R_1 \vee R_6$

So
$$W_6 = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \begin{matrix} R_1 \vee R_6 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \end{matrix}$$

Hence R^* is given by the matrix W_6.

$\therefore \quad R^* = \{(a_1, a_2), (a_1, a_3), (a_1, a_4), (a_1, a_5), (a_1, a_6), (a_1, a_2), (a_6, a_3), (a_6, a_5)\}$. ■

Theorem 16 (Most Important)

Let $W_k = \left[w_{ij}^{(k)}\right]$ be the relation matrix $\mathfrak{R}_k$.

Let $W_{k-1} = \left[w_{ij}^{(k-1)}\right]$,

Then $$w_{ij}^{(k-1)} = w_{ij}^{(k-1)} \vee \left(w_{ik}^{(k-1)} \wedge w_{kj}^{(k-1)}\right)$$

Proof: $w_{ij}^{(k)} = 1$ iff there exists an edge sequence from *ai* to *aj* whose interior vertices are from $\{a_1, a_2, a_3, \ldots, a_{k-1}\}$. There exist two cases *viz*.

Case I: The interior vertices of the edge sequence are from $\{a_1, a_2,..., a_{k-1}\}$.

This case arises iff $w_{ij}^{(k-1)} = 1$.

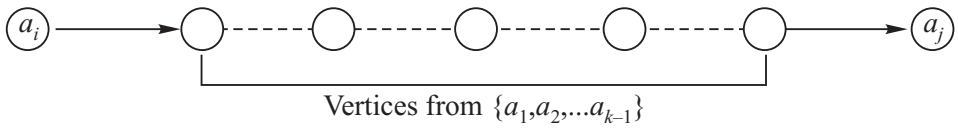


Fig. 4.12.

Case II: a_k is an interior vertex of the edge sequence.In this case the edge sequence can be split into two parts an edge sequence from a_i to a_k and an edge sequence from a_k to a_j. We could assume without loss of generality that a_k appears only once as an interior vertex. Then the interior vertices of edge sequence from a_i to a_k and a_k to a_j will be only from $\{a_1, a_2,..., a_{k-1}\}$.

Hence $w_{ik}^{(k-1)} = 1$ and $w_{kj}^{(k-1)} = 1$.

The case II arises iff $w_{ik}^{(k-1)} \wedge w_{kj}^{k-1} = 1$.

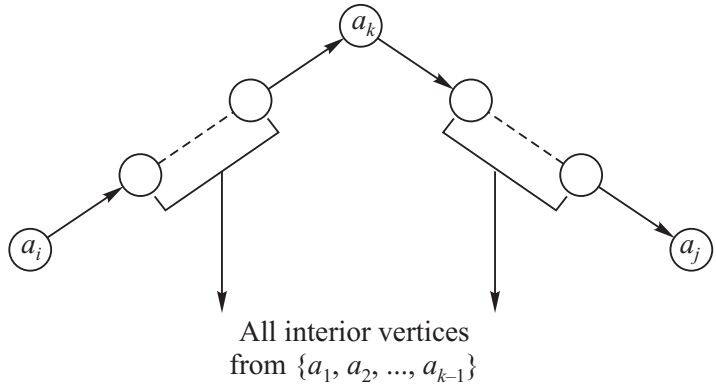


Fig. 4.13

Thus $w_{ij}^{(k)}$ iff either $w_{ij}^{(k-1)} = 1$ or $w_{ik}^{(k-1)} \wedge w_{kj}^{(k-1)} = 1$ **Hence Proved.**

4.17 Partial Order Relation

A set whose members can be composed in such a way is said to be ordered and specification of how its members compare with each other is called an ordered relation on the set.

Let $\mathbb{S}$ be a nonempty set. A relation $\mathbb{R}$ on $\mathbb{S}$ is said to be partial order relation if $\mathbb{R}$ is **reflexive**, **anti-symmetric** and **transitive** (*RAT*). A relation of partial order is often denoted by $\preceq$. *A* non empty set $\mathbb{S}$ together with a

relation of partial order $\preceq$ on $\mathbb{S}$ is called a poset (Partial order set) and is denoted by $(\mathbb{S}, \preceq)$

For example: $x \preceq y$ is a poset where means '$x \preceq y$' or $x, y \in \mathbb{S}$, where $\mathbb{S}$ is a set of real numbers.

Because,

(*i*) $x \leq x$, *i.e.* $\preceq$ is reflexive.

(*ii*) If $x \leq y$ and $y \leq x \Rightarrow x = y$ *i.e.* antisymmetric.

(*iii*) If $x \leq y$ and $y \leq z$ then $x \leq y$ *i.e.* $\preceq$ is transitive.

Let $\sum = \{a, b\}$. Define a relation R on $\sum^*$ as: xRy if x is a prefix of y. Is R a partial order?

- Every word is a prefix of itself, so R is reflexive.
- Let xRy and yRx. Then $y = sx$ and $x = ty$ for some $s, t \in \sum^*$, so $x = t(sx) = (ts)\, x$.

  Consequently, $ts = \lambda$ and hence $t = s = \lambda$. So $x = y$ and the relation is antisymmetric.
- Suppose xRy and yRz. Then $y = sx$ and $z = ty$ for some $s,t \in \sum^*$. Therefore, $z = t(sx) = (ts)x$. Consequently xRz, and the relation is transitive.

Thus, R is a partial order on $\sum^*$ and $(\sum^*, R)$ is a poset.

4.17.1 Totally Ordered Set

If any two elements is a post are comparable, such a poset is called totally ordered or a linear order. The poset is then a totally ordered set or a linearly ordered set. We can notice that $\leq$ is a total order on $\mathbb{R}$, where as the divisibility relation is not a total order on N.

4.17.2 Lexicographic Order

Let $(A, \preceq_1)$ and $(B, \preceq_2)$ be two posets. Define a relation $\preceq$ on $A \times B$ as $(a, b) \preceq (a', b')$ if $a \preceq a'$, or $a = a'$ and $b \preceq_2 b'$. The relation $\preceq$, an extension of the alphabetic order, is the **lexicographic order.**

The lexicographic order is a partial order on $A \times B$. If A and B are totally ordered sets, so is $A \times B$. The lexicographic order can be extended to the cartesian product $A_1 \times A_2 \times \ldots \times A_n$ of n posets and n totally ordered sets.

For example: Let $\sum$ be a partially ordered alphabet with the partial order $\preceq$ and $\sum^n$ denote the set of words of length n over $\sum$. Since every word in $\sum^n$ can be considered an n-tuple, the lexicographic order on the cartesian product on n posets can be applied to $\sum^n$ also.

Let $x = a_1a_2..., a_n$ and $y = b_1\ b_2... b_n$ be any two elements in $\sum^n$. Then $x \prec y$ if:

- Either $a_1 \preceq b_1$, or
- An integer i exists such that $a_1 = b_1$, $a_2 = b_2$,..., $a_i = b_i$, and $a_{i+1} \prec b_{i+1}$.

In particular, let $\sum$ denote the English alphabet, a totally ordered set: $a \prec b \prec c \prec... \prec z$. Clearly, *computer* $\prec$ *demolish, compress* $\prec$ *computer, contend* $\prec$ *content,* and *content* $\prec$ *context.*

This lexicographic order can work for $\sum^*$ in a familiar way. Let x and y be any two words over $\sum$. Then $x \prec y$ in lexicographic order if one of two conditions holds.

- $x = \lambda$, the empty word.
- If $x = su$ and $y = sv$, where s denotes the longest common prefix of x and y, the first symbol in u precedes that in v in alphabetic order.

For example, *marathon* $\prec$ *marble, margin* $\prec$ *market, limber* $\prec$ *timber,* and *creation* $\prec$ *discretion.*

Helmut Hasse (1898–1979)

A noted number theorist and dedicate teacher H. Hasse was born in Kassel, Germany. His father was a judge. After the gymanasiums, he joined the navy. While in the navy in the Baltic he studied number theory and then mathematics at the university of Kiel. He left navy is December 1918 and completed his Ph. D is 1921 from Marburg. He opted teaching as a career in Kiel in 1922. Three years later he became a professor at Halle, then moved to Marburg, Göttingen, Berlin and finally Hamburg in 1950, where he remained until retirement in 1966 in various superior posts. Hasse was a member of several mathematical organizations and won many awards for his sincere services.

4.17.3 Hasse Diagrams

It is possible; at least in principle, to draw a diagram which shows a glance the order relation on a finite poset. Let $(S, \preceq)$ be a poset. Define a relation $\prec$ on S by '$a \prec b$ if and only if $a \leq b$ but $a \neq b$, $a, b \in$ S. Given a partial order $\leq$ on S, b is said to cover a if $a < b$ and there is no element c in S such that $a \preceq c \preceq b$ holds.

A Hasse diagram of a postet $(S, \leq)$ is a graphical representation consisting of points labelled by the members of S, with a line segment directed generally upward from a to wherever b covers a.

SOLVED EXAMPLES

Example 1

Construct the Hasse diagram for the poset (A, 1) where A = {1, 2, 3, 6, 8, 12} and 1 *denotes the divisibility relation.*

Solution

The digraph of the poset is

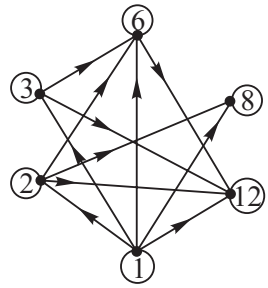


Fig. 4.14

Step 1: Delete the loop at each vertex. The result is as under:

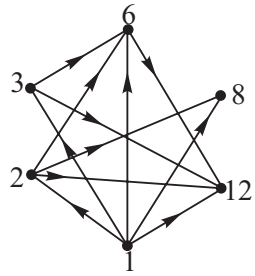


Fig. 4.15

Step 2: Delete all edges implied by transitivity and ensuing diagram below:

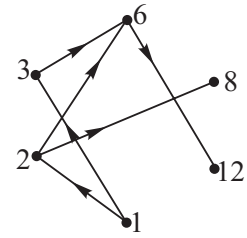


Fig. 4.16

Step 3: Omit all arrows and draw the edges 'upwards'. The Hasse diagram is

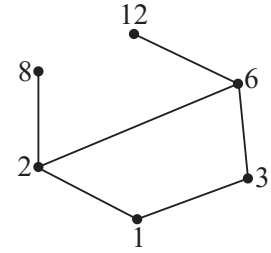


Fig. 4.17 ■

Example 2

Draw the Hasse diagram for the poset (A, ⊆), where A denotes the power set of the set {a, b, c}.

Solution

The set {*a*, *b*, *c*} has eight subsets:

Φ, {*a*}, {*b*}, {*c*}, {*a*, *b*}, {*b*, *c*}, {*a*, *c*}, and {*a*, *b*, *c*} following steps as example 1 above, we find the Hasse diagram as

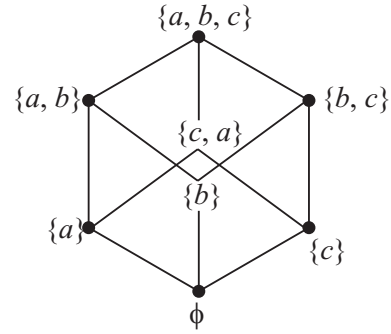


Fig. 4.13 ■

Example 3

Draw a Hasse diagram for (S, ≤) with six elements.

Solution

Let $S = \{1, 2, ,3, 4, 5, 6\}$. We define ≤ as $m \leq n$ if 'm divides n', that is, n is an integer multiple of m. The diagram in Figure 4.10 is a Hasse diagram of the poset $(S, \leq)$.

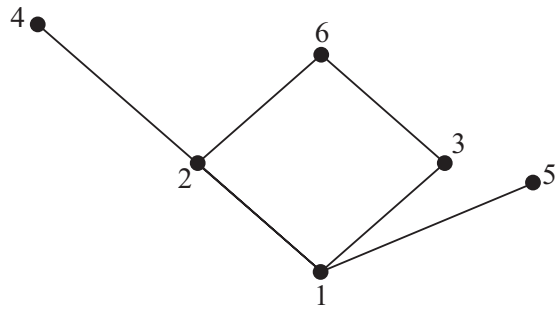


Fig. 4.17

There is no segment between 1 and 6 because 6 does not cover 1. From diagram we see that 2 covers 1 and 6 covers 2. Similarly, 4 does not cover 1 because 4 covers 2 and 2 covers 1. That is $1 < 2 < 4$.

■

SUMMARY

1. **Boolean Matrix**
   - A boolean matrix has bits for entries
   - The join $A \vee B$ and **meet** $A \vee B$ of two boolean matrices A and B are obtained by *oring* and *anding* the corresponding bits, respectively.
   - The **boolean product** $A \odot B$ two boolean matrices $A = (a_{ij})_{m \times p}$ and $B = (b_{jk})_{p \times n}$ where $c_{ij} = (a_{i1} \wedge b_{1j}) \vee (a_{i2} \wedge b_{2j}) \vee \ldots (a_{ip} \wedge b_{pj})$
   - **The complement** A' of a boolean matrix A results from swapping 0's and 1's
2. **Binary Relation**
   - A **binary relation** R from A to B is a subset of $A \times B$. If $(a, b) \in R$, we write ab

- A relation R from a finite set to finite set of a can be represented by its **adjacency matrix,** $\boldsymbol{M_R}$.
- A relation a finite set can by represented by **digraph**.
- Every function $f : A \to B$ is a binary relation from A to B such that (1) dom(f) = A; and (2) if $(a, b) \in f$ and $(a, c) \in f$, then $b = c$.

3. Properties of Relations

- A relation R on A is **reflexive** if aRa for every $a \in A$
- A relation R on A is **symmetric** if aRb implies bRa
- A relation R on A is **antisymmetric** if $aRb \wedge bRa$ implies $a = b$.
- A relation R on A is **transitive** if $aRb \wedge bRc$ implies aRc.

4. Constructing New Relations

- The **union** and **intersection** of two relations R and S from A to B are $R \cup S = \{(a, b) \mid aRb \vee aSb\}$; $R \vee S = \{(a, b \mid aRb \vee aSb\}$
- If R and S are relations on a finite set, $M_{R \cup S} = M_R \vee M_S$ and $M_{R \cap S} = M_R \wedge M_S$.
- Let R be a relation from B to C. Their **compositions** is $R \odot S = \{(a, c) \in A \times C \mid aRb \vee bRc$ for some b in $B\}$.
- In particular, if A, B and C are finite sets, then $M_{R \odot S} = M_R \odot M_S$
- For a relation R on a finite set, $M_{R^n} = (M_R)^{|n|}$
- For a transitive relation R, $R^n \subseteq R$ for every $n \geq 1$
- The **connectivity relation** R^∞ is the union of all powers of R:

$$R^\infty = \bigcup_{n=1}^{\infty} R^n; \; M_{R^\infty} = M_R \vee M_{R^2} \vee M_{R^3} \vee \ldots.$$

- In particular, let R be a relation on set with size n. Then

$$R^\infty = \bigcup_{i=1}^{n} R^i \text{ and } M_R \vee M_{R^2} \vee \ldots \vee M_{R^n}$$

5. Transitive Closure

- The **transitive closure** R^* of a relation R is the smallest transitve relation containing it.

- $R^* = R^\infty$
- **Warshall's algorithm** systematically finds M_{R^*}

6. **Equivalence Relations and Partitions**
   - An **equivalence relation** is reflexive, symmetric, and transitive.
   - An equivalence relation on a set induces a partition of the set and vice versa
7. **Partial and Total Orders**
   - A **partial order** $\leq$ is reflexive, antisymmetric, and transitive. A set together with a partial order is a **poset**
   - Two elements, x and y, in a poset are **comparable** if either $x \leq y$ or $y \leq x$
   - If any two elements in a poset are comparable, the partial order is a **total order** or **linear order**
   - The **Hasse diagram** of a finite poset contains no loops, edges implied by transitivity, or arrows; or arrows; its edges are drawn upward
   - The element of a finite nonempty poset can be **sorted topologically**

EXERCISE

1. Let $A = \{1, 2, 3, 4\}$ and $B = \{a, b, c\}$ and

$\mathcal{R} = \{(1, a), (1, b), (2, b), (2, c), (3, b), (4, a)\}$.

$\mathcal{S} = (1, b), (2, c), (3, b), (4, b)\}$

Compute $\overline{\mathcal{R}}$, $\mathcal{R} \cap \mathcal{S}$ and $\mathcal{R} \cup \mathcal{S}$.

[**Ans.** $\overline{\mathcal{R}} = \{(1, c), (2, a), (3, a), (3, c), (4, b), (4, c)\}$;
$\mathcal{R} \cap \mathcal{S} = \{(1, b), (3, b), (2, c);\}$ $\mathcal{R} \cup \mathcal{S} = \{(1, a),$
$(1, b), (2, b), (2, c), (3, b), (4, a), (4, b)]$

2. Let $A = B = \{a, b, c, d\}$,

$\mathcal{R} = \{(a, a), (a, c), (b , c), (c, a), (d, b), (d, d),$ and$\}$

$\mathbb{S} = \{(a, b), (b, c), (c, a)\ (c, b), (d, c)\}$ compute $M_{(\mathbb{R} \cap \mathbb{S})}$, $M_{(\mathbb{R} \cup \mathbb{S})}$, $M(\mathbb{R}^{-})$ and $M\,\mathbb{S}$.

$$\left[\begin{array}{l} \textbf{Ans.} = \quad M_{\mathbb{R} \cap \mathbb{S}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; M_{\mathbb{R} \cap \mathbb{S}} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \\ \\ (\mathbb{R}^{-1}) = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad M_{\overline{\mathbb{S}}} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix} \end{array}\right]$$

3. Let $\mathbb{R}$ be a relation on a set A. Prove the following:

(a) If $\mathbb{R}$ is reflexive, then $\mathbb{R} \cap \mathbb{R}^{-1} \neq \phi$

(b) If $\mathbb{R}$ is symmetric, then $\mathbb{R} \cap \mathbb{R}^{-1} \neq \phi$

(c) If $\mathbb{R}$ is antisymmetric, so is $\mathbb{R}^{-1}$.

4. Let A, B and C be infinite sets. Let $\mathbb{R}$ be a relation from A to B, and S a relation from B to C. Then Prove that

$M_{R \odot S}\, M_R \odot M_S$.

[**Hint:** Let $A = \{a_1, a_2, ..., a_m\}$, $B = \{b_1, b_2 ..., b_n\}$, and $C = \{c_1, c_2, ..., c_p\}$. Then the matrices M_R, M_S, $M_{R \odot S}$, and $M_R \odot M_S$ are of sizes $m \times n$, $n \times p$, $m \times p$ and $m \times p$, respectively.]

Let $M_{R \odot S} = (x_{ij})$ and $M_R \odot M_S$.

5. If R be a relation on a finite set and n be any positive integer. Then prove that $M_{R^n} = (M_R)^{|n|}$

6. If R be a transitive relation on a aet A. Then prove that $R^n \subseteq R$ for every positive integer n.

[Hint: Using PMI When $n = 1$, $R^1 \subseteq R$, which is true. Suppose $R^k \subseteq R$ for an arbitrary positive integer k.]

To show that $R^{k+1}. \subseteq R$:

Let $(x, y) \in R^{k+1}$. Since $R^{k+1} = R^k \odot R$, $(x, y) \in R^k \odot R$. Then, by definition, there is a z in A such that $(x, z) \in R^k$ and $(z, y) \in R$. But $R^k \subseteq R$, by the inductive hypothesis. Consequently, $(x, z) \in R$. Thus

$(x, z) \in R$ and $(z, y) \in R$, so $(x, y) \in R$ by transitivity. Thus $R^{k+1} \subseteq R$. Thus, by induction, $R^n \subseteq R$ for every $n \geq 1$.

7. Find the transitive closures of the relation $R = \{(a, b), (b, a), (b, c)\}$, $S = \{(a, a), (b , b), (c, c)\}$, and $T = \Phi$ on $\{a, b, c\}$.

[Hint:

- $R = \{(a, b), (b, c), (b, a)\}$. Since $(a, b) \in R$, and $(b, c) \in R$, it needs (a, c) to be transitive. So add (a, c) to R. The new relation is $R_1 = \{(a, b), (a, c), (b, c), (b, a)\}$. It contains both (a, b) and (b, a) but not (a, a) or (b, b). Add them to R_1: $R_2 = \{(a, a), (a, b), (a, c), (b, a), (b, b), (b, c)\}$. It is transitive and contains R, so it is the transitive closure of R.
- The relation S is transitive, by default, so $S^* = S$.
- The transitive closure of $\varnothing$ is itself.]

8. If $\sum$ denotes an alphabet. Define a relation R on $\sum^*$ by ${}_xRy$ if $||x|| = ||y||$ whose $||w||$ denotes the length of the word w. Is R an equivalence relation?

[Hint:

- Since every word has the same length as itself, R is reflexive.
- Suppose that xRy. Then $||x|| = ||y||$, so $||y|| = ||x||$. Consequently, yRx. Thus R is symmetric.
- If $xRy4$ and xRz, then $||x|| = ||y||$ and $||y|| = ||z||$. Therefore, $||x|| = ||z||$ and hence xRy. In other words, R is transitive. Thus, R is an equivalence relation.]

9. Find all equivalence classes of the congruence relation *mod 5* on the set of integers.

[Ans.

$[0] = \{..., -10, -5, 0, 5, 10,...\}$

$[1] = \{..., -9, -4, 1, 6, 11,...\}$

$[2] = \{..., -8, -3, 2, 7, 12,...\}$

$[3] = \{..., -7, -2, 3, 8, 13,...\}$

$[4] = \{..., -6, -1, 4, 9, 14,...\}]$

10. Prove that every finite nonempty poset $(A, \leq)$ has a minimal element.:

[**Hint:** Let a_1 be any element in A. If a_1 is not minimal, there must be an element a_2 in A such that $a_2 \prec a_1$. If a_2 is minimal, then we have finished. If a_2 is not minimal, A must have an element a_3 such that $a_3 \prec a_2$. If a_3 is not minimal, continue this procedure. Since A contains only a finite number of elements, it must terminate with some element a_n. Thus $a_n \prec a_{n-1} \prec ... \prec a_3 \prec a_2 \prec a_1$. Consequently, a_n is a minimal element.]

Suggested Readings

1. J.A. Gallian "*Assigning Driver's License Numbers*" Mathematics magazine, vol. 64 (feb. 1991), pp. 13-22.

2. T. Koshy, "*Elementary Number theory with Applications*", Harcourt/ Academic Press, Boston, MA, 2002, pp. 210-436.

3. T. Koshy, "*Discrete Mathematics with Applications*" Academic Press, Boston, MA, 2004.

4. N. Koblitz, "*Cryptography and Number Theory*" Springer, 2010.

5. S.K. Yadav, "*Elements of Graph Theory*", Ane books, New Delhi, 2011.

6. S.K. Yadav, "*Discrete Mathematical Structures*", Satya Prakashan, New Delhi, 2001.

Check for updates

5

Functions

Joseph Louis Lagrange (1736 – 1813)

Joseph Louis Lagrange (1736 – 1813) ranks with Leonhard Euler as one of the greatest mathematicians of the 18th century. The eldest of 11 children in a wealthy family in Turin, Italy, Lagrange was forced to pursue a profession often his father, an influential cabinet official, lost all his wealth by engaging in unsuccessful financial speculations.

While studying the classics at the College of Turin, the 17-year-old Lagrange found his interest in mathematics kindled by an essay by the astronomer Edumund Halley on the superiority of the analytical methods of calculus over geometry in the solution of optical problems. In 1754, he began corresponding with several outstanding mathematicians in Europe. The following year, he was appointed professor of mathematics at the Royal Artillery School in Turin. Three years later, he helped to found a society that later became the Turin Academy of Sciences. While at Turin, Lagrange developed revolutionary results in the calculus of variations, mechanics, sound, and probability, winning the prestigious Grand Prix of the Pairs Academy of Sciences in 1764 and 1766.

In 1766, when Euler left the Berlin Academy of Sciences, Frederick the Great wrote to Lagrange that ``the greatest king in Europe'' would like to have ``the greatest mathematician of Europe'' at his court. Accepting the invitation, Lagrange moved to Berlin to head the Academy and remained there for 20 years. When Frederick died in 1786, Lagrange moved to Pairs at the invitation of Louis XVI. He was appointed professor at the Ékcole Normale and then at the École Polytechnique, where he taught until 1799. He died in Pairs. Lagrange made significant contributions to analysis, analytic mechanics, calculus, probability, and number theory, as well ads helping to establish the French metric system.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_5

5.1 Introduction

The concept of a function is so fundamental that plays an important role of a unifying thread to intertwines every branch of mathematical sciences. It is almost used is our daily life as well. When we compute our electricity bill or water bill, we use the concept of functions unknowingly.

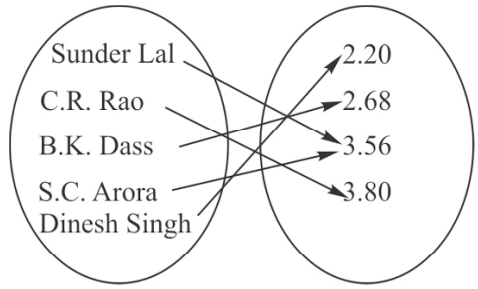


Fig. 5.1

Here is an example of a function from academic world. Let us consider five professors of mathematics majors – Sunder Lal, *C. R.* Ras, *B. K.* Dass, *S. C.* Arora and Dinesh Singh. Their quality point average (*QPA*) on a 0–4 scale are 3.56, 3.80, 2.68, 3.56 and 2.20 respectively. Each element in the set A = {Sunder Lal, *C. R.* Rao, *B. K.* Dass, *S. C.* Arora, Dinesh singh} is assigned a unique element from the set B = {3.56, 3.80, 2.68, 2.20}.

This assignmemt has two interesting properties:

(*i*) Every major is assigned a *QPA*

(*ii*) Every major has a unique *QPA*. Such assignment is a function.

Definition: Let X and Y be any two nonempty sets. A **Function** from X to Y is a rule that assign to each element $x \in X$ a unique element $y \in Y$. Functions are usually denoted by the letters *f, g, h, i* etc. If f is a function from X to Y we write $f: X \to Y$. The set X is the **domain** of the function f and Y the **codomain** of f denoted by *dom (f)* and *codom (f)* respectively. If $X = Y$ then f is said to be a function on X.

Let $f: X \to Y$, so every element $x \in X$ is paired with unique element $y \in Y$ (as shown in Fig. 5.2 below). Then y is the value (or image) of the function f at x, denoted by $f(x)$, and x is the pre-image of y under f; y is also known as the output corresponding to the imput (or argument) x. Thus $y = f(x)$.

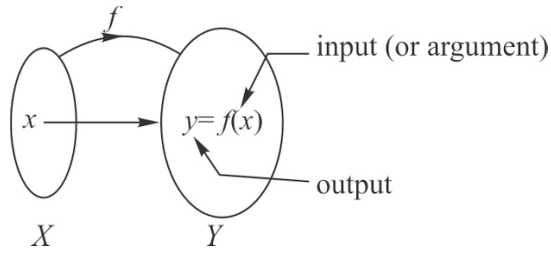


Fig. 5.2

Let $f: X \to Y$ and $A \subseteq X$. Then $f(A)$ denotes the set $\{f(a) : a \in A\}$ Particularly, $f(X)$ if the **range** of f, denoted by range (f). Thus range (f) = $f(X) = \{f(x) : x \in X\}$ Notice that range (f) $\subseteq Y$.

Note: The geometrical representation of a function is called a **graph**.

5.1.1 Sum and Product of Functions

Let $f: X \to \mathbb{R}$ and $g: Y \to \mathbb{R}$. There can be combined to construct new functions. The sum and product of f and g, denoted by $f + g$ and fg respectively, are defined as.

$$(f + g)(x) = f(x) + g(x)$$

$$(fg)(x) = f(x) \cdot g(x)$$

The functions $f + g$ *and* fg are defined whenever both f and g are defined.

Thus dom $(f + g)$ = dom (fg) = dom $(f) \cap$ dom (g)

SOLVED EXAMPLES

Example 1

A town in Massachusetts charges each household a minimum of $ 75 for up to 4000 cubic feet (ft^3) of water every 6 months. In addition, each household has to pay 60¢ for every 100 ft^3 of water in excess of 4000 ft^3. Express the water bill f (x) as a function of the number of cubic feet of water x used for 6 months.

Solution

The minimum charge is $ 75 for up to 4000 ft^3 of water, so $f(x) = 75$ if $0 \leq x \leq x$ 4000.

Suppose you used more than 4000 ft^3 of water. Then

$$\text{Cost for the excess} = \frac{x - 4000}{100}(0.60) = 0.006\,(x - 4000)$$

Then

Total cost = minimum charge + cost for the excess = $75 + 0.006\,(x - 4000)$

Thus, the water bill $f(x)$ in dollars can be computed using the piecewise definition:

$$f(x) = \begin{cases} 75 & \text{if } 0 \leq x \leq 4000 \\ 75 + 0.006(x - 4000) & \text{if } x > 4000 \end{cases}$$ ■

Example 2

Graph each function.

1. Let f: $\mathbf{Z} \to \mathbf{Z}$ defined by $f(x) = x^2$.

2. Let g: $\mathbb{R} \to \mathbb{R}$ defined by

$$g(x) = \begin{cases} 75 & \text{if } x \geq 0 \\ -1 & \text{if } -2 \leq x < 0 \\ 3x + 4 & \text{otherwise} \end{cases}$$

Solution

The graph of the functions are displayed in Figures 5.3 and 5.4 respectively. Notice that the graph of *f* is a discrete collection of points

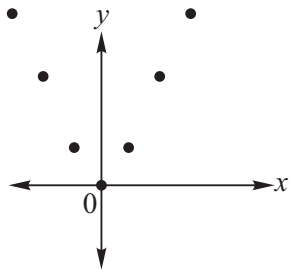


Fig. 5.3

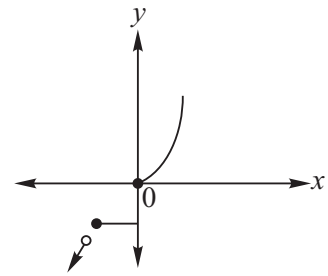


Fig. 5.4

■

Example 3

Determine whether or not each of the following relation is a function with domain {1, 2, 3, 4}. If any relation is not a function, explain why.

(a) $R_1 = \{(1, 1), (2, 1), (3, 1), (4, 1), (3, 3)\}$

(b) $R_2 = \{(1, 2), (2, 3), (4, 2)\}$

(c) $R_3 = \{(1, 1), (2, 1), (3, 1), (4, 1)\}$

(d) $R_4 = \{(1, 4), (2, 3), (3, 2), (4, 1)\}$

Solution

(*a*) R_1 is not a function, since there are 2 pairs (3, 1) and (3, 3), which means that the image of the element 3 is not unique.

(*b*) R_2 is not a function, since there is no image for the element 3 of the domain.

(*c*) R_3 is a function, even though the images of 1, 2, 3, 4 of the domain are one and the same element 1.

(*d*) R_4 is a function. ■

Example 4

Determine whether or not each of the following relations is a function. If a relation is a function, find its range.

(a) $R_1 = \{(x, y) | x, y \in Z, y = x^2 + 7\}$, *which is a relation from Z to Z.*

(b) $R_2 = \{(x, y) | x, y \in R, y^2 = x\}$, *which is a relation from R to R.*

(c) $R_3 = \{(x, y) | x, y \in R, y = 3x + 1\}$, *which is a relation from R to R.*

(d) $R_4 = \{(x, y) | x, y \in Q, x^2 + y^2 = 1\}$, *which is a relation from Q to Q.*

Solution

(*a*) R_1 is a function, since for each $x \in Z$ there is a unique y, given by $y = x^2 + 7$.

Range = {7, 8, 11, 16, 23,...}

(*b*) R_2 is not a function, since for a given x, the value of y is not unique as $= \pm \sqrt{x}$.

(*c*) R_3 is a function, since for each $x \in R$, there is a unique y, given $y = 3x + 1$.

When x is real, y is real.

Hence range = R, the set of real numbers.

(*d*) R_4 is not a function, since for a given x, the value of y is not unique, as $y = \pm \sqrt{1 - x^2}$. ■

5.2 Special Types of Functions

Let f be a function from A to B. Then say that f is **everywhere defined** if Dom (f) = A. We say that f is **onto** is Ran (f) = B. Finally we say that f is **one to one** if we cannot have $f(a) = f(a')$ for two distinct elements a and a' of A. The definition of one to one may be restated in the following equivalent form

If $f(a) = f(a')$, then $a = a'$

For example: Let $A = \{1, 2, 3, 4\}$ and $B = \{a, b, c, d\}$ and let $f = \{(1, a), (2, a), (3, d), (4, c)\}$.

Since Dom (f) A, f is everywhere defined. On the other hand, Ran (f) = $\{a, c, d\} \neq B$;

Therefore f is not onto.

Since $f(1) = f(2) = a$

We can conclude that f is not one-to-one.

If f: $A \rightarrow B$ is **one-to-one** function, then f associates to each element a of Dom (f) an element $b = f(a)$ of Ran (f). Every b in Ron (f) is matched, in thisway, with one and only one element of Dom (f). For this reason, such on f is often called a *bijection* between Dom (f) and Ran (f). If f is also everywhere defined and onto, then f is called *a one-to-one correspondence between A and B.*

(*i*) Into Mappings: A mapping $f: A \to B$ is said to be an into mapping if $f(A)$ is a proper subset of B. In this case we say that f maps A into B.

For example: Let $f: Z \to Z$ (the set of integers) be defined by $f(x) = 2x$, $x \in Z$. Then f is an into mapping, because $f(Z)$ is a proper subset of the codomain set Z.

(*ii*) Onto Mapping: A mapping $f: A \to B$ is said to be an onto mapping if $f(A) = B$. In this case we say that f maps onto B.

For example: Let $f: Z \to Z$ be defined by $f(x) = x + 1$, $x \in Z$. Then every element y in the codomain set Z has a pre-image $y - 1$ in the domain set Z. Therefore, $f(z) = Z$ and f is an onto mapping.

(*iii*) Injective Mapping: A mapping $f: A \to B$ is said to be injective (or one-to-one or injection) if for each pair of distinct elements of A their f-images are distinct.

Thus $f: A \to B$ is injective if $x_1 \neq x_2$ in A implies $f(x_1)$ in B.

For example: Let $f: R \to R$ be defined by $f(x) = 2x + 1$, $x \in R$. Then for $x_1, x_2 \in R$ $(x_1 \neq x_2)$ we have $f(x_1) \neq f(x_2)$.

(*iv*) Surjective Mapping: A mapping $f: A \to B$ is said to be surjective (or onto or surjective) if $f(A) = B$.

(*v*) Bijective Mapping: A mapping $f: A \to B$ is said to be bijective (or bijection) if f is both injective and surjective.

For example: Let $f: Z \to Z$ be defined by $f(x) = x + 2$, $x \in Z$. The mapping f is both injective and surjective. So it is bijective.

Note. If f is injective, each element of B has at most one pre-image. If f is surjective, each element of B has at least one pre-image. If f is bijective, each element of B has exactly one pre-image.

(*vi*) Many one Mapping: A mapping $f: A \to A$ is said to be many-one if two or more distinct elements in A have the same image. That is, if $f(x) = f(x)'$ implies $x \neq x'$.

(*vii*) Constant Mapping: A function f of A into B is called a constant mapping if the same element $b \in B$ is assigned to every element in A. In other words, $f: A \to B$ is a constant mapping if the range of f consists of only one element.

For example. Let $f: R \to R$ be defined by the formula $f(x) = 50$. Then f is a constant function since 50 is assigned to every element.

(*viii*) Identity Mapping: Let A be a set. Let the function $f: A \to A$ be defined by the formula $f(x) = x$. Then f is called the identity function or the identity mapping or the identity transformation on A.

(*ix*) Inverse Mapping: Let $f: A \to B$ be a one-one onto mapping. Then the mapping $f^{-1}: B \to A$ which associates to each element $b \in B$ the element $a \in A$, such that $f(a) = b$, is called the inverse mapping of the mapping $f: A \to B$.

Hence if $f: A \to B$ be a one-one on to mapping, then $f^{-1}: B \to A$. The mapping f^{-1} is called the inverse mapping of the mapping f.

For example. Let $A = R - (3)$ and $B = R - (1)$ and the function $f: A \to B$ be defined by

$f(x) = \dfrac{x-2}{x-3}$. Then the inverse function $f^{-1}: B \to A$ is given by

$$y = \frac{x-2}{x-3} \quad \text{or,} \quad x = \frac{2-3y}{1-y}$$

Hence $$f^{-1}(x) = \frac{2-3x}{1-x}.$$

5.2.1 Polynomial Function

A function $f: \mathbb{R} \to \mathbb{R}$ defined by $f(x) = a_n x^n + a_{n-1}x^{n-1} + ... + a_1 + a_0$, where $a_0, a_1, ..., a_n \in \mathbb{R}$, $n \in \boldsymbol{W}$, and $a_n \neq 0$, is a **polynomial function.** The expression $a_n x^n + a_{n-1}x^{n-1} + ... + a_1 x + a_0$ is a **polynomial of degree** n in x. When $n = 1$, f is a **linear function**; when $n = 2$, f is a **quadratic function.**

5.2.2 Exponential and Logarithmic Function

Let $a \in \mathbb{R}^+$, $a \neq 1$, and x any real number. The function $f: \mathbb{R} \to \mathbb{R}^+$ defined by $f(x) = a^x$ is an **exponential function** with base a. The most frequently used base in computer science is two. Figure 5.5 shows the graph of the exponential function $f(x) = 2^x$.

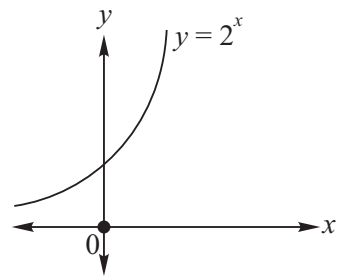


Fig. 5.5

Let $a \in \mathbb{R}^+$, $a \neq 1$, and x and y any real numbers such that $y = a^x$. Then x is called the **logarithm** of y to the base a, denoted by $\log_a y$. Thus $(\log_a y = x) \leftrightarrow (y = a^x)$. Accordingly, the

function $f: \mathbb{R}^+ \to \mathbb{R}$ defined by $f(x) = \log_a x$ is the **logarithmic function** with base a.

Remember that the most commonly used base in computer science is two. The corresponding logarithm is denoted by lg. Thus $\log x = \log_2 x$, and Figure 5.6 shows the graph of the logarithmic function $f(x) = \log x$.

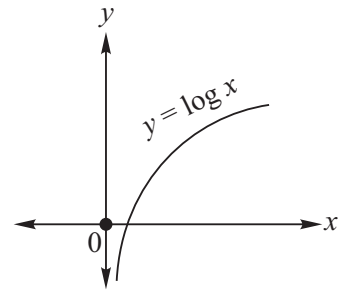


Fig 5.6

5.2.3 Floor and Ceiling Functions

The floor and ceiling functions are often used in the study of algorithms, as will be seen in the next two chapters. The **floor** of a real number x, denoted by $\lfloor x \rfloor$, is the greatest integer $\leq x$. The **ceiling** of x, denoted by $\lceil x \rceil$, is the least integer $\geq x$. The floor of x *rounds down* x while the ceiling of x *rounds up*. Accordingly, if $x \notin \mathbf{Z}$, the floor of x is the nearest integer to the *left of* x on the number line and the ceiling of x is the nearest integer to the *right of* x, as shows in Figure 5.7. The **floor function** $f(x) = \lfloor x \rfloor$ and the **ceiling function** $g(x) = \lceil x \rceil$ also known as the **greatest integer function** and the **least integer function**, respectively.

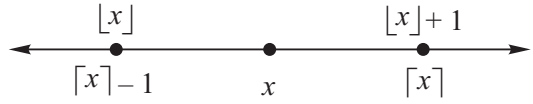


Fig. 5.7

For example, $\lfloor \pi \rfloor = 3$, $\lfloor \log 3 \rfloor = 1$, $\lfloor -3.5 \rfloor = -4$, $\lfloor -2.7 \rfloor = -3$, $\lceil \pi \rceil = 4$, $\lceil \log 3 \rceil = 2$, $\lceil -3.5 \rceil = -3$, $\lceil -2.7 \rceil = -2$, and $\lceil 3 \rceil = 3 = \lceil 3 \rceil$.

SOLVED EXAMPLES

Example 1

Find the number of positive integers ≤ 3000 and not divisible by 7 or 8.

Solution

Ler A $\{x \in \mathbb{N} \mid x \leq 3000$ and divisible by $7\}$ and $B = \{x \in \mathbb{N} \mid x \leq 3000$ and divisible by $8\}$. We need to find $|A' \cap B'|$:

$$\begin{aligned}|A' \cap B'| &= |(A \cup B)'| \\ &= |U| - |A \cup B| \\ &= |U| - |A| - |B| + |A \cap B| \\ &= 3000 - [3000/7] - [3000/8] + [3000/56] \\ &= 3000 - 428 - 375 + 53 = 2250\end{aligned}$$ ■

Example 2

Find the number of positive integers ≤ 2076 and divisible by 3, 5, or 7.

Solution

Let A, B, and C denote the sets of positive integers ≤ 2076 and divisible by 3, 5, and 7, respectively. By the inclusion-exclusion principle,

$$\begin{aligned}|A \cup B \cup C| &= |A| + |B| + |C| - |A \cap B| - |B \cap C| \\ &\quad - |C \cap A| + |A \cap B \cap C| \\ &= \left\lfloor \frac{2076}{3} \right\rfloor + \left\lfloor \frac{2076}{5} \right\rfloor + \left\lfloor \frac{2076}{7} \right\rfloor - \left\lfloor \frac{2076}{15} \right\rfloor \\ &\quad - \left\lfloor \frac{2076}{35} \right\rfloor - \left\lfloor \frac{2076}{21} \right\rfloor + \left\lfloor \frac{2076}{105} \right\rfloor \\ &= 692 + 415 + 296 - 138 - 59 - 98 + 19 = 1127\end{aligned}$$

■

Example 3

Prove that the number of leap years ℓ after 1600 and not exceeding a given year y is given by

$$\ell = \left\lfloor \frac{y}{4} \right\rfloor - \left\lfloor \frac{y}{100} \right\rfloor + \left\lfloor \frac{y}{400} \right\rfloor - 388$$

Solution

Let n be a year such that $1600 < n \le y$. To derive the formula for ℓ we proceed step-by-step:

- *To find the number of years n in the range divisible by 4:*

  Let $4n_1$ be such a year. Then $1600 < 4n_1 \le y$; that is, $400 < n_1 \le \frac{y}{4}$.

  Therefore, there are $n_1 = \left\lfloor \frac{y}{4} \right\rfloor - 400$ such years.

- *To find the number of centuries in the range* $1600 < n \le y$:
  Let $100n_2$ be a century such that $1600 < 100n_2 \le y$.
  Then $16 < n_2 \le \frac{y}{100}$.
  Therefore, there are $n_2 = \left\lfloor \frac{y}{100} \right\rfloor - 16$ centuries beyond 1600 and $\le y$.
- *To find the number of centuries in the range divisible by 400:*
  Since they are of the form $400n_3$, we have $1600 < 400n_3 \le y$. Then $4 < n_3 \le \frac{y}{400}$ so $n_3 = \left\lfloor \frac{y}{400} \right\rfloor - 4$.
  • Therefore,

$$\begin{aligned} \ell &= n_1 - n_2 + n_3 \\ &= \left\lfloor \frac{y}{4} \right\rfloor - 400 - \left\lfloor \frac{y}{100} \right\rfloor + 16 \left\lfloor \frac{y}{400} \right\rfloor - 4 \\ &= \left\lfloor \frac{y}{4} \right\rfloor - \left\lfloor \frac{y}{100} \right\rfloor - \left\lfloor \frac{y}{400} \right\rfloor - 388 \end{aligned}$$

■

5.2.4 Transcedental Function

A function which is not algebraic is called a transcedental function. *e.g.* The circular function, inverse circular function, exponential function, logarithmic function, hyperbolic function and inverse hyperbolic function are all transcedental functions.

5.2.5 Identity Function

The function $f: A \to A$ where $f(x) = x$, where $x \in A$ is called the **identity function** on A.

In other words, the identity function is the function that assigns each element of A to itself and is denoted by IA or simply I. The function IA is a bijection.

5.2.6 Integer Value and Absolute Value Functions

The integer value of x, where x is a real number, converts x into an integer by truncating or deleting the fractional part of the number and denoted by ***INT* (*x*).**

e.g. $INT\,(2.35) = 3$, $INT\,(-8.54) = -8$ and $INT\,(9) = 9$ etc.

5.2.7 Remainder Function

If a is any integer and m a positive integer, then the integer remainder (function) when a is divided by m is denoted by $a \pmod m$. [read as 'a modulo m'] *viz*, $a \pmod m$ is the unique integer r *s.t.* $a = mq + r$. where $0 \le r$ *c* m. When a is positive, we simply divide a by m to get remainder r. *e.g.* $30 \pmod 5 = 0$, $3 \pmod 5 = 3$, $30 \pmod 7 = 2$ when a is negative, we divide $|a|$ by m and get the remainder r'.

Then $a \pmod m = m - r'$ $(r' \neq 0)$ *e.g.* $-30 \pmod 5 = 0$, $-30 \pmod 7 = 7 - 2 = 5$ and $-3 \pmod 5 = 2$ when two integer a and b have the same remainder when divided by the positive integer m, then a is said to be congruent to b modulo m and denoted as

$$a \equiv b \pmod m$$

when $a \equiv b \pmod m$, m divides $a - b$ or $a - b$ is a multiple of m and hence $a = b + k\,m$.

If $a \equiv b \pmod m$ and $c \equiv d \pmod m$, then (*i*) $a + c \equiv (b + d) \pmod m$ and (*ii*) $ac \equiv (bd) \pmod m$.

Since, $a \equiv b \pmod m$ and $c \equiv d \pmod m$, we have $a = b + k_1 m$ and $c = d + k_2 m$.

$\therefore$ $a + c = b + d + (k_1 + k_2)m = (b + d) + km$, say.

$\therefore$ $a + c = (b + d) \pmod m$

Now $ac = (b + k_1 m)(d + k_2 m)$

$= bd + (bk_2 + dk_1 + k_1 k_2 m)m$

$= bd + k'm$, say.

$\therefore$ $ac = (bd) \pmod m$.

For example, $16 \equiv 2 \pmod 7$ and $10 \equiv 3 \pmod 7$. Hence, $26 \equiv 5 \pmod 7$ and $160 \equiv 6 \pmod 7$

5.3 Composition of Functions

If f: $A \to B$ and g: $B \to C$, then the composition of f and g is a new function from A to C denoted by $g \circ f$, is given by.

$$(g \circ f)(x) = g\{f(x)\}, \forall\, x \in A.$$

To find $(g \circ f)(x)$, we find first the image of x under f and then find the image of $f(x)$ under g. Thus the range of f is the domain of g. Sometimes $g \circ f$ (read as 'g of f') is also called the relative product of the function f and g or left composition of g will f.

Note:
- If f and g are treated as relations, then the composition of f and g was denoted by $f \circ g$, whereas the composition of the functions f and g is denoted by $f \circ g$.
- When $g \circ f$ is defined, $f \circ g$ need not to be defined, for, when $g \circ f$ is defined $R_f = D_g$. This does not mean that $R_g = D_f$, which is the required condition for $f \circ g$ to exist *viz*. composition of function is not commutative.

Property 1: Composition of functions is associative *i.e.* if $f: A \to B$, $g: B \to C$ and $h: C \to D$ are functions, then, $h \circ (g \circ f) = (h \circ g) \circ f$.

Proof: We have $f: A \to B$ and $g: B \to C$,then $g \circ f: A \to C$

Since $g \circ f: A \to C$ and $h: C \to D$, then $h \circ (g \circ f): A \to D$

Now $f: A \to B$ and $h \circ g: B \to D$. Hence $(h \circ g) \circ f: A \to D$

Thus, the domain and codomain of $h \circ (g \circ f)$ and those of $(h \circ g) \circ f$ are the same.

Let $x \in A$, $y \in B$ and $z \in C$, so that

$$y = f(x) \text{ and } z = g(y).$$

Then $$(g \circ f)(x) = g\{f(x)\} = g(y) = z$$

$\therefore$ $$h \circ (g \circ f)(x) = h(z) \quad \ldots(1)$$

Also $$\{(h \circ g)\} \circ f\}(x) = (h \circ g)\{f(x)\}$$

$$= h[g\{f(x)\}] = h[g(y)] = h(z) \quad \ldots(2)$$

From (1) and (2) we get

$h \circ (g \circ f) = (h \circ g) \circ f$, since they are true for all $x \in A$.

Note: $h \circ (g \circ f)$ or $(h \circ g) \circ f$ may also be written as $h \circ g \circ f$.

Property 2: When $f: A \to B$ and $g: B \to C$ are functions, then $g \circ f: A \to C$ is an injection or bijection according as f and g are injections surjections or bijections.

Proof: (*i*) Let $a_1, a_2 \in A$.

Then $(g \circ f)(a_1) = (g \circ f)a_2 \Rightarrow g\{f(a_1)\} = g\{f(a_2)\}$

$\Rightarrow f(a_1) = f(a_2)$ ($\because$ g is injective)

$\Rightarrow a_1 = a_2$ ($\because$ f is injective)

$\therefore$ $g \circ f$ is injective.

(*ii*) Let $c \in C$.

Since g is onto, there is an element $b \in B$ such that $c = g(b)$.

Since f is onto, there is an element $a \in A$ such that $b = f(a)$.

Now $(g \circ f)(a) = g\{f(a)\} = g(b) = c$

This means that $g \circ f: A \to C$ is onto.

(*iii*) From (*i*) and (*ii*), it follows that $g \circ f$ is bijective when f and g are bijective.

5.4 Inverse of a Function

If $f: A \to B$ and g: $B \to A$, then the function g is called the *inverse* of the function f, iff $g \circ f = I_A$ and $f \circ g = I_B$.

If $x \in A$ and $y \in B$, then

$(g \circ f)(x) = I_A(x)\ g\{f(x)\} = x$ ...(*i*)

$(f \circ g)(y) = I_B(y)\ f\{g(y)\} = y$ ...(*ii*)

From (*i*) and (*ii*), we have, if $y = f(x)$, then $x = g(y)$ and so forth $\Rightarrow g: B \to A$ is inverse of $f: A \to B$, if $x = g(y)$ whenever $y = f(x)$

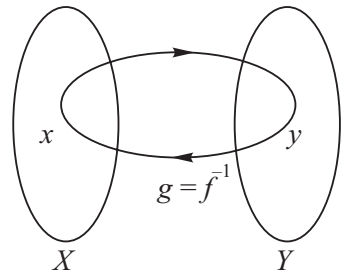


Fig. 5.7

The inverse of f, *viz.* g is denoted by f^{-1}.

Thus, if f^{-1} is the inverse of f, then $x = f^{-1}(y)$, whenever $y = f(x)$

Note: g^{-1} , the inverse of g is f.

Properties of Inverse of a Function

Property 1: The inverse of a function f, if exists, is unique.

Proof: If possible, let g and h be inverse of f.

Then by definition, $g \circ f = I_A$ and $f \circ g = I_B$ ...(1)

Also $h \circ f = I_A$ and $f \circ h = I_B$ ...(2)

Now $h = h \circ I_B = h \circ (f \circ g)$, by (1)

$= (h \circ f) \circ g$, by associativity

$= I_A \circ g$, by (2) $= g$

Hence the result.

Note: By the uniqueness property, we have $(f^{-1})^{-1} = g^{-1} = f$.

Property 2: The necessary and sufficient condition for the function $f: A \to B$ to be invertible (*viz.,* for f^{-1} to exist) is that f is one-to-one and onto.

Proof:

(*i*) Let $f: A \to B$ be invertible.

Then there exists a unique function $g: B \to A$ such that

$$g \circ f = I_A \text{ and } f \circ g = I_B \qquad ...(1)$$

Let $a_1, a_2 \in A$ such that $f(a_1) = f(a_2)$,

where $f(a_1), f(a_2) \in B$ [$\because f : A \to B$ is a function]

Since $g : B \to A$ is a function.

$$g\{f(a_1)\} = g\{f(a_2)\}$$

i.e., $(g \circ f)(a_1) = (g \circ f)(a_2)$

i.e., $I_A(a_1) = I_A(a_2)$, by (1)

i.e., $a_1 = a_2$

Thus, whenever $f(a_1) = f(a_2)$, we have

$$a_1 = a_2$$

Hence, f is one-to-one.

(*ii*) Let $b \in B$. Then $g(b) \in A$, since $g : B \to A$ is a function.

Now $b = I_B(b) = (f \circ g)(b)$, by (1)

$= f\{g(b)\}$

Thus, corresponding to every $b \in B$, there is an element $g(b) \in A$ such that $f\{g(b)\} = b$.

Hence f is onto.

Thus, the necessary part of the property is proved.

(*iii*) Let $f : A \to B$ be bijective.

Since, f is onto, for each $b \in B$ there exists an $a \in A$ such that $f(a) = b$.

Hence, we can define a function $g : B \to A$ by $g(b) = a$,

where $f(a) = b$ ...(2)

If possible, let $g(b) = a_1$ and $g(b) = a_2$ where $a_1 \neq a_2$.

This means that $f(a_1) = f(a_2) = b$, which is not possible, since f is one-to-one.

Thus, $g : B \to A$ is a unique function.

Also from (2), we get $g \circ f = I_A$ and $f \circ g = I_B$

i.e., f is invertible.

Thus the sufficient part of the property is proved.

Property 3: If $f : A \to B$ and $g : B \to C$ are invertible functions, then $g \circ f : A \to C$ is also invertible and $(g \circ f)^{-1} = f^{-1} \circ g^{-1}$

viz., the inverse of the composition of two functions is equal to the composition of the inverses of the functions in the reverse order.

Proof: Since $f : A \to B$ and $g : B \to C$ are invertible, they are bijective.

Hence, $(g \circ f) : A \to C$ is also bijective (by an earlier property)

$\therefore$ $g \circ f$ is also invertible. *viz.,* $(g \circ f) : C \to A$ exists

Since $g^{-1} : : C \to B$ and $f^{-1} : : B \to A, f^{-1} \circ g^{-1} : C \to A$ can be formed.

Thus both $(g \circ f)^{-1}$ and $f^{-1} \circ g^{-1}$ are functions from C to A.

Now for any $a \in A$, let $b = f(a)$ and $c = g(b)$ ...(1)

$\therefore$ $(g \circ f)(a) = g\{f(a)\} = g(b) = c$

$\therefore$ $(g \circ f)^{-1}(c) = a$ ...(2)

By the assumption (1), $a = f^{-1}(b)$ and $b = g^{-1}(c)$

$\therefore$ $(f^{-1} \circ g^{-1})(c) = f^{-1}\{g^{-1}(c)\} = f^{-1}(b) = a$ ...(3)

From (2) and (3), it follows that

$(f \circ g)^{-1} = f^{-1} \circ g^{-1}$, since f^{-1}, g^{-1} and $(g \circ f)^{-1}$ are bijective.

SOLVED EXAMPLES

Example 1

Give an example of a function $N \to N$ as a set of ordered pairs which is

(a) one-to-one but not onto,

(b) onto but not one-to-one,

(c) both one-to-one and onto,

(d) neither one-to-one nor onto.

Solution

(*a*) If $A = \{1, 2, 3\}$, $B = \{x, y, z, w\}$ and $f = \{(1, w), (2, x), (3, y)\}$, then the function f is one-to-one, as distinct elements of A are mapped into distinction elements of B. f is not onto, as the range of f is not B *viz.,* $z \in B$ is not the image of any element of A.

(*b*) If $A = \{1, 2, 3\}$, $B = \{x, y, z\}$ and $f = \{(1, x), (2, y), (3, z), (4, x)\}$, then the function f is onto, as the range of f is the entire B.

f is not one-to-one, since $1 \neq 4$, though $f(1) = f(4) = x$.

Note: If A and B are sets containing m and n elements respectively, no function $A \to B$ can be onto when $n > m$.

(*c*) If $A = \{1, 2, 3\}$, $B = \{x, y, z\}$ and $f = \{(1, z), (2, y), (3, x)\}$, then f is both one-to-one and onto, as $f(1), f(2), f(3)$ are all different and the range of f is B.

(*d*) If $A = \{1, 2, 3\}$, $B = \{x, y, z\}$ and $f = \{(1, x), (2, y), (3, x)\}$, then f is neither one-to-one nor onto, as $f(1) = f(3) = x$, but $1 \neq 3$ and as the range of f is not B. ■

Example 2

Determine whether each of the following functions is an injection and/or a surjection.

(a) $f : R \to R$, *defined by* $f(x) = 3x^3 + x$

(b) $f : Z^+ \to Z^+$, *defined by* $f(x) = x^2 + 2$

(c) $f : R \to R$, *defined by* $f(x) = -4x^2 + 12x - 9$

(d) $f : Z \to Z$, *defined by* $f(x) = x^2 + 14x - 51$.

Solution

(*a*) Let us consider the graph of $y = f(x)$, *i.e.,* $y = 3x^3 + x$, which is given in Fig. 5.6.

From the graph, it is obvious that for a given real value of x, there is only one real value of y, as a line drawn parallel to the y-axis intersects the graph at only one point. Hence, the function is injective.

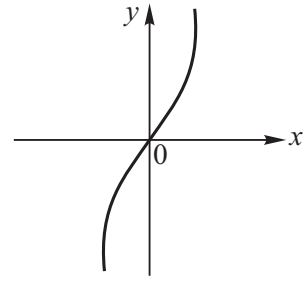


Fig. 5.8

Similarly for a given real value of y, there is only one real value of x, as a line drawn parallel to the x-axis intersects the graph at a point. Hence the function is surjective.

viz., the function $f(x) = 3x^2 + x$ is bijective.

(*b*) $$f(x) = x^2 + 2$$

$\therefore$ $$f(x_1) = f(x_2), \text{ if } x_1^2 + 2 = x_2^2 + 2$$

i.e., if $(x_1 - x_2)(x_1 + x_2) = 0$

i.e., if $x_1 = x_2$ ($\because x_1 + x_2 \neq 0$, as $x_1, x_2 \in z^+$)

$\therefore f(x)$ is injective.

When $y = f(x)$, *i.e.,* $y = x^2 + 2$, we have $x^2 = y - 2$

$\therefore$ When $y = 1$, x does not exist.

Also when $y = 4$, $x = \pm\sqrt{2} \notin z^+$

Hence $f(x)$ is not surjective.

(*c*) $$f(x) = -4x^2 + 12x - 9 = -(2x - 3)^2$$

We note that $f(1) = f(2) = -1$, but $1 \neq 2$.

Hence $f(x)$ is not injective.

If $$y = f(x) = -(2x - 3)^2, \text{ we have}$$

$$(2x - 3)^2 = -y \text{ or } x = \frac{1}{2}\left(3 \pm \sqrt{-y}\right)$$

Hence, for positive real values of y, x is not defined. Hence $f(x)$ is not surjective.

$\therefore f(x)$ is neither injective nor surjective.

(*d*) $$f(x) = x^2 + 14x - 51$$

If $$f(x_1) = f(x_2), \text{ we have}$$

$$x_1^2 + 14x_1 - 51 = x_2^2 + 14x_2 - 51$$

i.e., $$(x_1^2 - x_2^2) + 14(x_1 - x) = 0$$

i.e., $$(x_1 - x_2)(x_1 + x_2 + 14) = 0$$ ■

If we assume that $x_1 \neq x_2$, then $f(x_1)$ can be equal to $f(x_2)$, provided $x_1 + x_2 = -14$, for example, when $x_1 = -10$ and $x_2 = -4$.

Thus, when $x_1 = -10$ and $x_2 = -4$, *i.e.,* $x_1 \neq x_2$, $f(x_1) = f(x_2) = -91$.

$\therefore$ $f(x)$ is not injective.

If $$y = f(x) = x^2 + 14x - 51, \text{ we have}$$

$$y + 100 = (x + 7)^2.$$

$\therefore$ When $y \in Z$ and less than -100 (for example, $y = -101$), x does not exist.

Hence, $f(x)$ is not surjective.

i.e., $f(x)$ is neither injective nor surjective.

Example 3: Two Queens Puzzle

There are two queens on an 8×8 chessboard. One can capture the other if they are on the same row, column, or diagonal. The 64 squares on the

board are numbered 0 through 63. Suppose one queen is in square x and the other in square y, where $0 \le x, y \le 63$. Can one queen capture the other?

Since the squares are labeled 0 through 63, we can label each row with the numbers 0 through 7, and each column with the same numbers 0 through 7. In fact, each row label = $\lfloor r/8 \rfloor$ and each column label = c and mod 8, where $0 \le r, c \le 63$. See Figure 5.8. Thus, the queen in square x lies in row $\lfloor x/8 \rfloor$ and column x mod 8, and that in square y lies in row $[y/8]$ and column y mod 8. Consequently, the two queens will be in the same row if and only if $\lfloor x/8 \rfloor = \lfloor y/8 \rfloor$ and in the same column if and only if x mod 8 = y mod 8. For example, if $x = 41$ and $y = 47$, the two queens lie on the same row.

	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	8	9	10	11	12	13	14	15
2	16	17	18	19	20	21	22	23
3	24	25	26	27	28	29	30	31
4	32	33	34	35	36	37	38	39
5	40	41	42	43	44	45	46	47
6	48	49	50	51	52	53	54	55
7	56	57	58	59	60	61	62	63

← column label

↑
row
label

Fig. 5.8

How do we determine if they lie on the same diagonal? There are 15 northeast diagonals and 15 southeast diagonals. With a bit of patience, we can show that the queens lie on the same diagonal if and only if the absolute value of the difference of their row labels equals that of the difference of their column labels; that is, if and only if $|\lfloor x/8 \rfloor - \lfloor y/8 \rfloor| = |x \bmod 8 - y \bmod 8|$.

For example, let $x = 51$ and $y = 23$; see Figure. Then $|\lfloor 51/8 \rfloor - \lfloor 23/8 \rfloor| = |6 - 2| = 4 = |3 - 7| = |51 \bmod 8 - 23 \bmod 8|$, so one queen captures the other. On the other hand, if $x = 49$ and $y = 13$, then $|\lfloor 49/8 \rfloor - \lfloor 13/8 \rfloor| \neq |49 \bmod 8 - 13 \bmod 8|$; so one queen cannot capture the other.

5.5 HASHING FUNCTIONS

Suppose that we must maintain the customer records for a large company and will store the information as computer records. We begin by assigning each customer a unique 9-digit account number. A unique identifier for a record is called its key. For now we will not consider exactly how and what information will be stored for each customer account, but will describe only the storage of a location in the computer's memory where this information will be found. To determine to which list a particular record should be assigned, we create a *hashing function* from the set of keys to the set of list numbers. Hashing functions frequently use a mod function, as shown in the next example.

Banks use nine-digit account numbers to create and maintain customer accounts. Customer records are stored in an array in a computer and can be accessed fairly easily and quickly using their unique **keys**, which in this case are the account numbers. Access is often accomplished using the **hashing function** $h(x) = x$ and mod m, where x denotes the key (account number) and m the number of cells in the array; $h(x)$ denotes the **hash address** of the customer record with key x. See Figure. 5.9

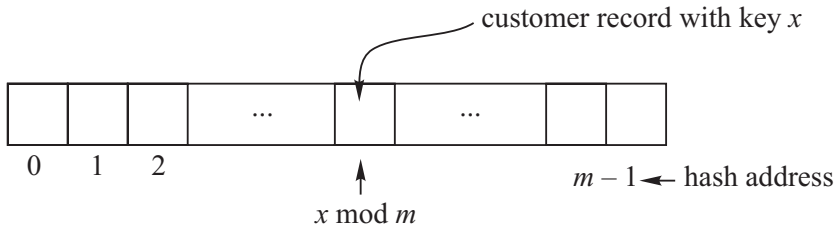


Fig. 5.9.

In particular, let $m = 1009$ and $x = 207630764$. The corresponding record is stored in location.

$$h(207630764) = 207630764 \bmod 1009 = 762$$

Likewise,

$$h(307620765) = 307620765 \bmod 1009 = 881$$

Since the hashing function is not injective (why?), theoretically different customer records can be assigned to the same location. For example,

$$h(207630764) = 762 = h(208801204)$$

This results in a **collision**.

One simple way to resolve a collision is to do a sequential search for the next available cell, beginning with the cell where the collision has occurred. Then we store the item in the available cell. If we come to the end of the array without any success, then we would continue the search back at the beginning of the array, as if the array were circular. This way of resolving a collision is called **linear probing**. So we would store the data with the account number 208801204 in location 763 (assuming that is available).

5.6 Countable and Uncountable Sets

A set S is countably infinite if there exists a bijection between S and $\mathbb{N}_0$. A set that is finite or countably infinite is called countable. A set that does not follow countably is uncountable.

The cardinality of $\mathbb{N}$ is denoted by $\mathbb{N}_0$ (read "aleph-naught," "aleph" being the first letter of the Hebrew alphabet. This symbol was introduced by Cantor). Thus a set S is countably infinite if $| S | = \aleph_0 = | \mathbb{N} |$.

If A and B are finite sets such that $A \subset B$, then $| A | < | B |$. This, however, need not be true in the case of infinite sets. For example, $E \subset \mathbb{N}$, where E denotes the set of even positive integers; nonetheless, $| E | = | \mathbb{N} | = \aleph_0$, as shown by the pairings in Figure.

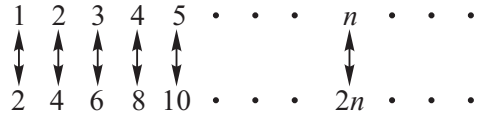


Fig. 4.10

SOLVED EXAMPLES

Example 1

Show that $N \times N$ is countably infinite.

Solution

Although we shall not give a formal proof, the arrows in Figure show how the various elements of $N \times N$ can be listed as the first, second, third, and so on in a systematic way, showing that $N \times N$ is countably infinite.

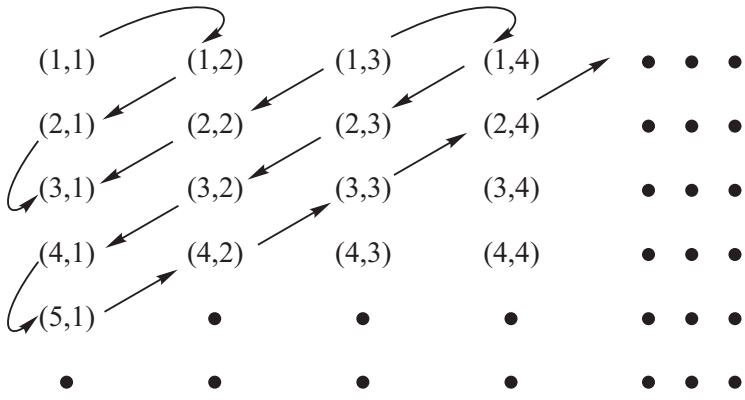


Fig. 4.11

■

Example 2

Show that the open interval (0, 1) is uncountable.

Solution (by contradiction)

Assume that the interval (0, 1) is countable. Then every real number between 0 and 1 can be listed as $a_1, a_2, a_3, \ldots$Each a_i has a unique decimal expansion (for numbers with two different decimal expansions, choose the expansion with trailing 9's. For example, although 0.5 = 0.5000… = 0.4999…, select 0.4999… for our discussion):

$$a_1 = 0.a_{11}a_{12}a_{13}a_{14}\ldots$$
$$a_2 = 0.a_{21}a_{22}a_{23}a_{24}\ldots$$
$$a_3 = 0.a_{31}a_{32}a_{33}a_{34}\ldots$$
$$a_4 = 0.a_{41}a_{42}a_{43}a_{44}\ldots$$
$$\vdots$$

where each a_{ij} is a digit.

Now construct a real number $b = 0.b_1b_2b_3b_4\ldots$as follows:

$$b_i = \begin{cases} 1 & \text{if } a_{ii} \neq 1 \\ 2 & \text{if } a_{ii} \neq 1 \end{cases}$$

Clearly, $0 < b < 1$; therefore b must be one of the numbers in the above list $a_1, a_2, a_3, a_4, \ldots$However, since $b_i \neq a_{ii}$ for every i, b cannot be in the list. This leads to a contradiction. Therefore, the real numbers between 0 and 1 cannot be listed and hence the interval (0, 1) is uncountable. (The technique employed is called **Cantor's diagonalization procedure**.) ■

5.7 Characteristic Function of A Set

Let E be a universal set and A be the subset of E. The function $\Psi_A : E \to \{0,1\}$ defined by

$$\Psi_A(x) = \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \notin A \end{cases}$$

is called the characteristic function of the set A.

Let us illustrate this by taking an example below:

Let E be a set of all people living in Jhunjhunu and let F be the set of all femals in this city. The Ψ_f associate the number 1 with each female and 0 with each male in Jhunjhunu. The following properties suggest how we can use the characteristic functions of sets to determine set relations.

Let A and B be any two subsets of a universal set E. The following holds for all $x \in E$. The following holds for all $x \in E$.

(*i*) $\Psi_A(x) = 0 \Leftrightarrow A = \phi$

(*ii*) $\Psi_A(x) = 1 \Leftrightarrow A = E$

(*iii*) $\Psi_A(x) \leq \Psi_B(x) \Leftrightarrow A \subseteq E$

(*iv*) $\Psi_A(x) = \Psi_B(x) \Leftrightarrow A = B$

(*v*) $\Psi_{A \cap B}(x) = \Psi_A(x) * \Psi_B(x)$

(*vi*) $\Psi_{A \cup B}(x) = \Psi_A(x) + \Psi_B(x) - \Psi_{A \cap B}(x)$

(*vii*) $\Psi_{\sim A}(x) = 1 - \Psi_A(x)$

(*viii*) $\Psi_{A \sim B}(x) = \Psi_{A \cap \sim B}(x) = \Psi_A(x) - \Psi_{A \cap B}(x)$

we can follow that

$x \in A \cap B \Leftrightarrow x \in A \cap x \in B$, so that $\Psi_A(x) = 1$ and $\Psi_B(x) = 1$ and $\Psi_{A \cap B}(x) = 1 * 1 = 1$

If $x \notin A \cap B$, then $\Psi_{A \cap B}(x) = 0$ and $\Psi_A(x) = 0$ or $\Psi_B(x) = 0$.

Consequently, $\Psi_A(x) * \Psi_B(x) = 0$.

Note: The operations $\leq$, $=$, $+$, $*$ and $-$ used with the characteristic functions are the usual arithmetic operations because the values of the characteristic functions are always either 0 or 1. On the other hand, the equality used for sets is a usual set equality. Other set operations used above are $\cup$, $\cap$, $\sim$ and $-$. The above properties can easily be proved using the definition of characteristic functions.

SOLVED EXAMPLES

Example 1

Show that $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

Solution

$$\begin{aligned}
\Psi_{A \cap (B \cup C)}(x) &= \Psi_A(x) * \Psi_{B \cup C}(x) && \text{(using prop. } v\text{)} \\
&= \Psi_A(x)\,(\Psi_B(x) + \Psi_C(x) - \Psi_{B \cap C}(x) && \text{(prop. } vi\text{)} \\
&= \Psi_A(x) * \Psi_B(x) + \Psi_A(x) * \Psi_C(x) - \Psi_A(x) * \Psi_{B \cap C}(x) \\
&= \Psi_{A \cap B}(x) + \Psi_{A \cap C}(x) - \Psi_{A \cap (B \cup C)}(x) && \text{(prop } v\text{)} \\
&= \Psi_{A \cap B}(x) + \Psi_{A \cap C}(x) - \Psi_{A \cap B \cap C}(x) \\
&= \Psi_{A \cap B}(x) + \Psi_{A \cap C}(x) - \Psi_{(A \cap B) \cap (A \cap C)}(x) \\
&= \Psi_{(A \cap B) \cap (A \cap C)}(x) && \text{(prop } vi\text{)}
\end{aligned}$$

■

Example 2

Show that $\sim(\sim A) = A$

Solution

$$\begin{aligned}
\Psi_{\sim \sim A}(x) &= 1 - \Psi_{\sim A}(x) \\
&= 1 - (1 - \Psi_A(x)) && \text{(Prop. } viii\text{)} \\
&= \Psi_A(x)
\end{aligned}$$

■

Example 3

Using characteristic functions, prove that

$$\begin{aligned}
A \cup (B \cap C) &= (A \cup B) \cap (A \cup C) \\
&= \Psi_{A \cup B}(x) \,.\, \Psi_{A \cup C}(x) = \{\Psi_A(x) + \Psi_B(x) - \Psi_{A \cap B}(x)\} \\
&\quad \times \{\Psi_A(x) + \Psi_C(x) - \Psi_{A \cap C}(x)\} \\
&= \{\Psi_A(x) \,.\, \Psi_A(x) + \Psi_A(x) \,.\, \Psi_C(x) \\
&\quad - \Psi_A(x) \,.\, \Psi_{A \cap C}(x)\} + \{\Psi_B(x) \,.\, \Psi_A(x) \\
&\quad + \Psi_B(x) \,.\, \Psi_C(x) - \Psi_B(x) \,.\, \Psi_{A \cap C}(x)\} \\
&\quad - \{\Psi_{A \cap B}(x) \,.\, \Psi_A(x) + \Psi_{A \cap B}(x) \,.\, \Psi_C(x) \\
&\quad - \Psi_{A \cap B}(x) \,.\, \Psi_{A \cap C}(x)\}
\end{aligned}$$

$$= \{\Psi_A(x) + \Psi_{A\cap C}(x) - \Psi_{A\cap C}(x) + \Psi_{A\cap B}(x)$$
$$+ \Psi_{B\cap C}(x) - \Psi_{A\cap B\cap C}(x) - \Psi_{A\cap B}(x)$$
$$+ \Psi_{A\cap B\cap C}(x) - \Psi_{A\cap B\cap C}(x)\}, \quad \text{by property}$$
$$= \Psi_A(x) + \Psi_{B\cap C}(x) - \Psi_{A\cap B\cap C}(x)$$
$$= \Psi_{A\cap(B\cap C)}(x),$$

Hence the result. ■

Example 4

Using characteristic functions, prove that

$\Psi_{A\oplus B}(x) = \Psi_A(x) + \Psi_B(x) - 2\Psi_A(x) \,.\, \Psi_B(x)$

$\Psi_{A\oplus B}(x) = 1 \Leftrightarrow x \in A \oplus B$

$\Leftrightarrow x \in A$ and $x \notin B$ or $x \notin A$ and $x \in B$

$\Leftrightarrow \Psi_x(A) = 1$ and $\Psi_x(B) = 0$ or $\Psi_x(A) = 0$ and $\Psi_x(B) = 1$

$\Leftrightarrow \Psi_x(A) + \Psi_x(B) - 2\Psi_x(A) \,.\, \Psi_x(B) = 1$

$\therefore \Psi_{A\oplus B}(x) = \Psi_x(A) + \Psi_x(B) - 2\Psi_x(A) \,.\, \Psi_x(B)$, when $x \in A \oplus B$ …(1)

Now $\Psi_{A\oplus B}(x) = 0 \Leftrightarrow x \in A \oplus B$

$\Leftrightarrow x \notin A \cap \overline{B}$ and $x \notin \overline{A} \cap B$

$\Leftrightarrow \Psi_{A\cap\overline{B}}(x) = 0$ and $\psi_{A\cap B}(x) = 0$

$\Leftrightarrow \Psi_A(x) \,.\, \psi_{\overline{B}}(x) = 0$ and $\Psi_{\overline{A}} \,.\, \Psi_B(x) = 0$

$\Leftrightarrow \Psi_A(x)\ \{1 - \Psi_B(x)\} = 0$ and $\{1 - \Psi_A(x)\}\ \Psi_B(x) = 0$

$\Leftrightarrow \Psi_A(x) = 0$ or $\Psi_B(x) = 1$ and $\Psi_A(x) = 1$ or $\Psi_B(x) = 0$

$\Leftrightarrow \Psi_A(x) = 0$ and $\Psi_B(x) = 0$ and $\Psi_A(x) = 1$ and $\Psi_B(x) = 1$

$\Leftrightarrow \Psi_A(x) + \Psi_B(x) - 2\Psi_A(x) \,.\, \Psi_B(x) = 0$, when $x \notin A \oplus B$ …(2)

From (1) and (2), we get the required result. ■

5.8 Permutation Function

The set of all one-to-one onto functions from A to A called the set of *permutation functions* from A to A.

For example if $A = \{1, 2, 3\}$, there are $3! = 6$ bijective functions from A to A, which are given below.

$$f_1 = \{(1, 1), (2, 2), (3, 3)\}; f_2 = \{(1, 1), (2, 3), (3, 2)\}$$
$$f_3 = \{(1, 2), (2, 3), (3, 1)\}; f_4 = \{(1, 2), (2, 1), (3, 3)\}$$
$$f_5 = \{(1, 3), (2, 1), (3, 2)\}; f_6 = \{(1, 3), (2, 2), (3, 1)\}$$

The images of {1, 2, 3} under the functions $f_1, f_2, \ldots f_6$ are obtained by permutations of {1, 2, 3}.

The set of functions $f_1, f_2, \ldots f_6$ denote by F is the set of permutation functions from {1, 2, 3} to {1, 2, 3}.

The compositions of the elements of F are given in the following composition table:

•	f_1	f_2	f_3	f_4	f_5	f_6
f_1	f_1	f_2	f_3	f_4	f_5	f_6
f_2	f_2	f_1	f_6	f_5	f_4	f_3
f_3	f_3	f_4	f_5	f_6	f_1	f_2
f_4	f_4	f_3	f_2	f_1	f_6	f_5
f_5	f_5	f_6	f_1	f_2	f_3	f_4
f_6	f_6	f_5	f_4	f_3	f_2	f_1

Also we note that $f_1^{-1} = f_1; f_2^{-1} = f_2; f_3^{-1} = f_5; f_4^{-1} = f_4; f_5^{-1} = f_3$ and $f_6^{-1} = f_6$.

If the set A has n elements, there are $n!$ elements in the set of permutation functions from A to A.

5.9 Growth of Functions

Let f and g be the functions whose domains are sets of some positive integers. We can say that f is $O(g)$, which is read as "***f* is big-oh of *g***", if there exist constant c and k s.t.

$$|f(n)| \leq C\,|g(n)| \;\forall\; n \leq k.$$

Then, we can say that f grows no faster than g does. If f is $O(g)$ and g is $O(f)$, we can say that f and g have the same order.

If f is $O(g)$ but g is not $O(f)$, we can say that f is of lower order than g or that f grows more slowly than g.

5.10 The Relation Θ

Let f and g be two functions having domains as the sets of some positive integers. We can say that f is related to g by a relation Θ *i.e.,* $f \Theta g$ iff f and g have the same order.

The relation Θ is an equivalent relation

We know that every function has the same order as itself. The relation Θ is reflexive.

The definition of $f \Theta g$ itself indicates that $g \Theta f$ whenever $f \Theta g$ and vice-versa *i.e.,* the relation is symmetric.

We can prove Θ is transitive. To do so, we take f, g, h such that $f \Theta g$ and $g \Theta h$. Then f and g have the same order and g and h have the same.

SOLVED EXAMPLES

Example 1

Show that the functions f and g defined by $f(x) = 3n^4 - 5n^2$ and $g(n) = n^4$, where n is a positive integer and have the same order.

Solution

$$\begin{aligned} f(n) &= 3n^4 - 5n^2 \le 3n^4 + 5n^2 \\ &\le 3n^4 + 5n^4, \forall\ n \ge 1. \\ &= 8n^4 = 8g(n) \ \forall\ n \ge 1. \end{aligned}$$

$$\therefore \quad |f(n)| \le 8\,|g(n)| \ \forall\ n \ge 1$$

$$|f(n)| \le c\,|g(n)| \ \forall\ n \ge k,$$

where $c = 8$ and $k = 1$. Hence f is $O(g)$

Conversely, we note that

$$\begin{aligned} g(n) &= n^4 = 3n^4 - 2n^4 \le 3n^4 - 5n^2, \forall\ n \ge 2 \\ &= f(n) \ \forall\ n \ge -2 \end{aligned}$$

(because if $n \ge 2$, then $n^2 > \frac{5}{2}$, $2n^2 > 5$ and $2n^4 > 5n^2$.)

$$\therefore \quad |g(n)| \le c\,|f(n)| \ \forall\ n \ge k$$

where $c = 1$ and $k = 2$. Hence g is $O(f)$

Since f is $O(g)$ and g is $O(f)$, if follows that f and g have the same order. ■

Example 2

Consider the functions

and

$$f(x) = \begin{cases} 2^n & \text{if } n \text{ is an even integer} \\ n & \text{otherwise} \end{cases}$$

$$f(x) = \begin{cases} 2^n & \text{if } n \text{ is an odd integer} \\ n & \text{otherwise.} \end{cases}$$

Prove that f(n) and g(n) are not of the same order.

Solution

Assuring the $f(n)$ is $O(g(n))$. For any c and k, we have $f(n) \leq c.\ g(n)\ \forall\ n \geq k$.

When n is even, this means that $2^n \leq c.\ n$ which is not true. Therefore, our assumption is wrong.

Hence, $f(n)$ is not $Q((g(n))$

Similarly, when n is odd, we find that $g(n)$ is not $O(f(n))$. This proves that $f(n)$ and $g(n)$ do not have the same order. ■

Example 3

Prove that every logarthmic function f (n) = $\log_b n$ has the same order as g(n) = $\log_2 n$.

Solution

Let us recall the identity

$$\log_b n = \frac{\log_a n}{\log_a b} \qquad \text{(taking } a = 2\text{)}$$

$$|\log_b n| \leq \frac{1}{\log_2 b} |\log_2 n|$$

and

$$|\log_2 n| \leq \log_2 b.\ |\log_b n|$$

These show that f is $O(g)$ and g is $O(f)$.

This show that f and g have the same order.

This shows that there exists pairs.

(c_1, k_1), (c_2, k_2), (c_3, k_3) and (c_4, k_4) s.t.

$$|f(n)| \leq c_1 |g(n)| \ \forall\ n \geq k_1;$$

$$|g(n)| \leq c_2 |f(n)| \ \forall\ n \geq k_2;$$

$$| g(n) | \leq c_3 | h(n) | \forall\ n \geq k_3;$$
$$| g(n) | \leq c_4 | g(n) | \forall\ n \geq k_4;$$

These relations yields:

(*a*) $| f(n) | \leq c_1(c_3 | h(n) |)$ if $n \geq k_1$ and $n \geq k_3$.
$\leq c_5 | h(n) | \forall\ n \geq k_5;$
where $c_5 = c_1\ c_3$ and k_5 is maximum of k_1 and k_3.

(*b*) $| h(n) | \leq c_6 | f(n) | \forall\ n \geq k_6 \leq c_4(c_2 | f(n) |)$ if $n \geq k_4$ and $n \geq k_2$ where $c_6 = c_4\ c_2$ and k_5 is maximum of k_4 and k_2.

Results (*a*) and (*b*) show that f and h have the same order *i.e.,* $f \Theta h$.

Thus, if $f \Theta g$ and $g \Theta h$, then $f \Theta h$.

Accordingly, the relations Θ is transitive hence the relation Θ is equivalent relation. ■

Rules for determining the Θ class of a function

(*i*) Θ (1) functions are constant and have zero growth, the slowest growth possible.

(*ii*) Θ ($l\ g\ n$) is lower than Θ (n^k) if $k > 0$.

(*iii*) Θ (n^a) is lower than Θ (n^b) iff $a < b$.

(*iv*) Q (a^n) is lower than Θ (b^n) iff $a < b$.

(*v*) Θ (n^k) is lower than Θ (a^n) for any n^k and any $a > 1$.

(*vi*) If $r \neq 0$, then Θ (r_f) = Θ (f) for any function f.

(*vii*) If h is a non-zero function and Θ (f) is lower than $\Theta(g)$ then $\Theta(fh)$ is lower than $\Theta(gh)$

(*viii*) If Θ (f) is lower than Θ (g), then Θ ($f + g$) = Θ (g).

Deductions: (*i*) Since Θ is an equivalence relation, we may think of equivalence classes of Θ. These classes consist of functions that have the same order.

e.g. $\Theta(1)$ is the class of functions of same order as $f(n) = 1$. *i.e.* constant function.

(*ii*) Θ (n) is the class of functions whose orders are equal to order of the function $f(n) = n$.

(*iii*) Θ (lgn) is the class of functions whose orders are equal to the order of the function $f(n) = (\log n)$ [$\log n$ means $\log_2 n$]

Example 4

Consider the functions f and g defined by $f(n) = \frac{1}{2}(n^3 + n^2)$ *and* $g(n) = n^3$*, where n is a positive integer. Prove that f is O(g).*

Solution

Let us note that

$$f(n) = \frac{1}{2}(n^3 + n^2) \leq \frac{1}{2} n^3 + \frac{1}{2} n^3$$

$$= n^3 = g(n) \; \forall \; n \geq 1.$$

i.e. $f(n) \leq g(n) \; \forall \; n \geq 1 \ldots$

or $|f(n)| \leq c \, |g(n)| \; \forall \; n \geq k.$

where $c = 1$ and $k = 1$.

Accordingly, f is $O(g)$.

i.e. $\frac{1}{2}(n^3 + n^2)$ is $O(n^3)$. ■

Example 5

Prove that if n is a non-negative integer, then the Θ *class of 2n is lower than the* Θ *class of 3n.*

Solution

We note that $2^n \leq c3^n \; \forall \; n \geq 0$, where $c = 1$.

Therefore, 2^n is $0(3^n)$

Assume that 2^n is $0(2^n)$.

There exist c and k s.t. $\forall \; n \geq k$, $3^n \leq c.\, 2^n$.

or $(3/2)^n \leq c$. or

$$n \leq \log_{3/2} c = \frac{\log_e c}{\log_e (3/2)}$$

This cannot hold for all $n \geq 0$.

Therefore, 3^n is not $0(2^n)$.

2^n is of lower order than 3^n. This implies that the Θ class of 2^n is lower than the Θ class of 3^n. ■

Example 6

Without using the Rules for Θ classes, prove that the polynomial function $p(n) = a_0 + a_1 n + a_2 n_2 + \ldots + a_m n^2$ *is* $0(n^m)$.

Solution

Let $b_0 = |a_0|, b_1 = |a_1|, \ldots b_n = |a_n|$. Then

$$|p(n)| \leq b_0 + b_1 n + b_2 n^2 + \ldots + b_m n^m.$$

$$= \left(\frac{b_0}{n^m} + \frac{b_1}{n^{m-1}} + \ldots + b_m \right) n^m$$

$$\leq (b_0 + b_1 + \ldots + b_m)\, n_m.$$

Thus, if we set $C = b_0 + b_1 + \ldots + b_m$, then we have

$|p(n)| \leq C\,|n_m| \;\forall\; n \geq 1$.

i.e. $p(n)$ is $0(n^m)$. ■

SUMMARY

1. **Function**
   - A function $f: X \to Y$ is a pairing of every element x in X with a unique element y in Y. Dom $(f) = X$, comod $(f) = Y$, and range $(f) = \{f(x) \in | \; x \in X\}$.
2. **Special Functions**
   - Polynomial function $f(x) = \sum_{l=0}^{n} a_i x^i \; (a_n \neq 0)$.
   - Exponential function $f(x) = a^x \; (a > 0, a \neq 1)$.
   - Logarithmic function $f(x) = \log_a x \; (a > 0, a \neq 1)$.
   - Absolute value function $f(x) = |x|$.
   - Floor function $f(x) = \lfloor x \rfloor$.
   - Ceiling function $f(x) = \lceil x \rceil$.
   - Characteristic function $f_A(x) = \begin{cases} 1 \; if \; x \in A \\ 0 \text{ otherwise} \end{cases}$.

- Mod function $f(x, y) = x \bmod y$.
- Div function $g(x, y) = x \text{ div } y$.

3. Properties of Functions

- A function $f : X \to X$ is the **identity function** on X if $f(x) = x$ for every $x \in X$.
- $f : X \to Y$ is **injective,** if $x_1 \neq x_2 \to f(x_1) \neq f(x_2)$.
- $f : X \to Y$ is **surjective,** if range $(f) = Y$.
- $f : X \to Y$ is **bijective,** if it is both injective and surjective.
- If X and Y are finite sets with $|X| = |Y|, f : X \to Y$ is injective if and only if it surjective.
- Two sets have the same cardinality if and only if a bijection exists between them.

4. Composition

- The **composition** of the functions $f : X \to Y$ and $g : Y \to Z$ is given by $(g \circ f)(x) = g(f(x))$ for every x in X.
- The composition of two bijections is bijective.
- The function $g : Y \to X$ is the inverse of the function $f : X \to Y$ if $g \circ f = 1_X$ and $f \circ g = 1_Y$
- A function is invertible if and only if it is bijective.

5. Sequences and the Summation Notation

- A sequences $\{s_n\}$ is a function with domain $X = \{a, a + 1, a + 2, ...\}$ of a finite subset of X, where $a \in \mathbf{W}$.
- Using the summation symbol Σ, the sum $a_k + a_k + 1 + \cdots + a_m$ is written as $\sum_{i=k}^{i=m} a_i$.

EXERCISE

1. Let $A = B = \{1, 2, 3, 4\}$, $f = \{(1, 4), (2, 1), (3, 2), (4, 3)\}$, and $g = \{(1, 2), (2, 3), (3, 4), (4, 1)\}$.

Verify that f and g are invertible and that $g = f'$ and $f = g^{-1}$

2. Let $A = \{x : x \text{ is real and } x \geq -1\}$, and $B = \{x : x \text{ is real, } x \geq 0\}$ consider $f : A \to B$ defined by $f(a) = \sqrt{a+1}$, $\forall\ a \in A$, show that f is invertible and determine f^{-1}. [**Ans.** $f^{-1}(b) = b^2 - 1.\ \forall\ b \in B.$]

3. For the functions f and g defined by $f(x) = x^3$ and $g(x) = x^2 + 1\ \forall\ x \in R$ find gof and fog. [**Ans.** gof $= x^6 + 1, fog = (x^2 + 1)^3$]

4. If $f : A \to B, g : B \to C$ and $h : C \to E$ be three functions. Prove that $(hog)of = ho(gof)$.

5. Let $f : A \to B$ and $g : B \to C$ be invertible functions. Then prove that gof is invertible and $(gof)^{-1} = f^{-1}\text{og}^{-1}$.

6. Suppose $f : A \to B$ and $g : B \to C$ are one-to-one functions. Show that $gof : A \to C$ is one-to-one.

7. Given $f : A \to B$ and $g : B \to C$. Show that if gof is one-to-one, then f is one-to-one.

8. Let $P = \{Ai\}$ be a partition of a set S. (a) Define the natural map f from S to P. (b) Prove that the natural map $f : S \to P$ is an onto function.

9. Plote the graph of $f(x)$ if $f(x) = [x] - [x]$
[**Hint:** Here $f(x) = 0$ if x is an integer and $f(x) = 1$ otherwise]

10. Simplify (a) $\dfrac{n!}{(n-1)!}$ (b) $\dfrac{(n+2)!}{n!}$

11. Suppose $P_{(n)} = a_0 + a_1 n + a_2 n_2 + \ldots + a_m n_m$; *i.e.,* suppose degree $P_{(n)} = m$. Prove that $P_{(n)} = 0_{(nm)}$.

12. Prove that if n is a non-negative integer, then the Θ class of 2^n is lower than the Θ class of 3^n.

Suggested Readings

1. D.I.A. Cohen. *Basic Techniques of Combinatorial Theory*, Wiley, New York, 1978, pp. 144–178.

2. R.L. Graham et al., *Concrete Mathematics,* Addison—Wesley, Reading, MA, 1989, pp 397–428.

3. F.S. Koltz, "*When is a Program Like a Function*?" Mathematics Teacher. Vol. 79 (Nov. 1986), pp. 648–651.

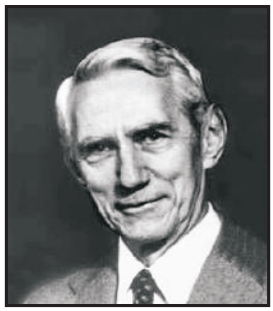
4. Thomas Koshy, *Discrete Mathematics with Applications*, Academic Press, California, USA, 2010.

5. S.K. Yadav, *Discrete Mathematical Structures*, Satya Prakashan, New Delhi, 2001.

❑❑❑

6

Lattice Theory



Claude Elwood Shannon (1916 – 2001)

A pioneer in artificial intelligence and information theory Dr. C.E.Shannon was born in Gaylord, Michigan. After his graduation from the university of Michigan in 1936, he received his masters in 1937 succeeding Ph.D. in applied mathematics is 1940 from MIT.

Shannon joined as research mathematician at the Bell Telephone Labs in 1941 and left in 1957 until he become professor of electrical engineering at MIT in 1958. In 1948, he coined the word bit for binary digit. He received honorary doctorates from many universities and number of honours and awards including the Nobel Peace Prize, MLM award, and IEE award. Shannon made outstanding contributions to the swiching and circuit theory, information security and cryplography, Information theory and artificial intelligence.

6.1 Introduction

A lattice is defined as a system of elements with two basic operations *viz*

(*i*) Formation of meet.

(*ii*) Formation of Joint.

with notations:

Meet of elements *a*, *b*	**Join of elements *a*, *b***
$a \cap b$	$a \cup b$
$a \wedge b$	$a \vee b$
$a \cdot b$ or ab	$a + b$.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_6

The first symbols are very much used, when the elements are sets, but they are different to read, easy to confuse and lack standard verbal equivalence. The writers would suggest readings "a et b" for $a \cap b$, "a vel b" for $a \cup b$. The Latin monosyllables are genuine connections, could not be shorter and exactly characterised meet and join of elements is propositional logic.

The second notation is much the same as the first but, curiously enough, is easier to read.

The third notation has all the advantages of simplicity and familiarity.

e.g. $a(b + c) = ab + ac$. is much perspicuous than

$$a \cap (b \cap c) = (a \cap b) \cup (a \cap c)$$

this distributive formula implies

$$a \cup (b \cup c) = (a \cup b) \cap (a \cup c)$$

which, written $a + bc = (a + b)\,(a + c)$ might appear bizarre to the unwary student.

6.2 Partial Ordered Sets

As we have already been introduced to the concepts of relations and ordering in the set theory. Let us recall that a reletion is a set S is simply a specified subset R of the cartesian product $S \times S$. Thus, it is a collection of some ordered pairs (a, b) of elements of S. If a certain pair (a, b) is in R, we think of a and b as somehow related and vice-versa. Thus the statements $(a, b) \in R$ and ${}_a R_b$ are equivalent.

A POSET $(X, \leq)$ is a set X together with a partial order relation $\leq$ on X *i.e.,*a relation which is **reflexive**, **antisymmetric** and **transitive**. (In short we may define it as RAT)

Specially we need

(i)	$a \leq a, \forall\ a \in X$	(Reflexivity)
(ii)	$a \leq b,\ b \leq a \Rightarrow a = b$	(Antisymmetry)
(iii)	$a \leq b,\ b \leq c \Rightarrow a \leq c$	(Transitivity)

We may observe that these three conditions do not necessarily and imply that given any two elements a and b, either $a \leq b$ or $b \leq a$ holds. Thus there may be such elements, which are not comparable. Hence the name partial order. Observing the conditions (i) and (ii) may be combined as

$$a \leq b,\ b \leq a \Leftrightarrow a = b.$$

Those posets in which this additional condition.

(*iv*) Given $a, b \in X$, either $a \leq b$ or $b \leq a$ holds, are called **totally ordered** or **linearly ordered** sets or **chains.**

6.2.1 Some Important Terms

The following are some important terms used in lattice theory:

(*a*) Usual Order: The number sets are in usual order.

N = all natural numbers

Z = all integers

Q = all rational numbers

R = all real numbers.

The natural order $\leq$ (less than or equal to) strict order $<$ (strictly less than)

and $\leq = < U =$ etc.

$\geq$ is also a partial (total) order.

(*b*) Set Inclusion: Let $X = P(S)$, the power set *i.e.,* the set of all subsets of a given set S. We may order X by, set inclusion or set containment: For $A, B \in X$, *i.e.,* $A, B, \subseteq S$.

$A \subseteq B$ iff $(X\, A \Rightarrow X \in B)$.

then clearly $A \subseteq A\ \forall\ A \subseteq S;\ A \subseteq B, B \subseteq\ X \Rightarrow A = B$;

and $A \subseteq B, B \subseteq C \Rightarrow A \subseteq C;$ for $A, B, C \in\ X$;

Thus $(X, \leq)$ is a poset. It is not totally ordered if S has more than one elements. For example if a, b, c, are distinct elements of S, then (a) and (b) or (a, b) and (b, c) are not comparable.

(*c*) Integer Divisibility: Take $X = N$, the set of all positive integers under the relation of "divides". Thus we say that a divides b *i.e.*, $(a \mid b)$ if b is an integral multiple of a *i.e.,* $\exists\ C \in N$ such that $b = a \cdot c$. It may be readily checked that the three conditions hold. But observe that on the set of all integers, the above relation is not a partial order as a and $(-a)$ both divide each other without being equal. Thus the relation as well as the set matter.

(*d*) Statement Algebra: The set of all statements (or that of all the statement formulas in a certain number of variables) forms a poset under the ordering of "**tautological implication**".

$P \leq Q$ iff $P \Rightarrow Q$ tautologically.

Here of course we are identifying tautologically equivalent statements, otherwise Antisymmetry would not hold.

(*e*) Subalgebraic Structures: The set of all subgroups of a given group or that of all subspace of a given space from poset under the relation of set inclusion.

(*f*) Lexicographic Order: Let $(X, \leq)$ be a poset. We may order the Cartesian product $X \times X$ by

$$(a, b) \leq (c, b) \text{ if } a \leq c \text{ or } a = c, b \leq d.$$

Then again $X \times X$ becomes a partially ordered set under this relation. This is called the lexicographic order as the words in a dictionary are ordered according to the above principle. The above order readily extends to a cartesian product of any number n of copies of X.

$$(a_1, ..., a_n) \leq (b_1, ..., b_n) \text{ if } a \leq i \leq n - 1$$

such that $a_1 = b_1 \; \forall \; 1 \; i$ and $a_{i+1} \leq b_{i+1} + 1$.

And then taking the union of these Cartesian products we got the set of all (open ended) finite strings of elements of X on which the above order extends.

$$a_1, a_2, ... a_h, \leq b_1 b_2, ... b_i$$

if $\exists \; i$ such that $a_i = b_i \; \forall \; j \leq i$ and $a_{i+1} \leq b_{i+1}$.

Observe that the dictionary uses this order but only on the set of all valid words with X as the alphabet set ordered.

$a \leq b \leq c$... Thus abacus ≤ algae.

(*g*) Divisors of a given positive integer: Consider the set $D(n)$ of all positive divisors of a positive integer n under the relation of "divides". This gives a poset for the same reason as the third example above and is in fact contained in that. But observe that depending upon n, this poset may or may not be totally ordered *e.g.*

$D(81) = \{1, 3, 9, 27, 81\}$ is totally ordered has $1 \mid 3 \mid 9 \mid 27 \mid 81$ but

$D(12) = \{1, 2, 3, 4, 6, 12\}$ is not totally ordered as 4 and 6 do not divide each other. Can you identify the numbers n for which $D(n)$ is totally ordered?

6.2.2 Diagramatical Representation of a Poset (Hasse Diagram)

Each poset can be represented by the help of a diagram. To draw the diagram of a poset, we represent each element by a small circle (a dot) and any two comparable elements are joined by lines in such a way that if $a \leq b$ then a lies below b in the diagram. Non-comparable elements are not joined. Thus there will not be any horizontal lines in the diagram of a poset.

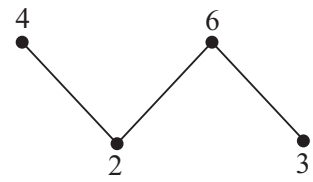


Fig. 6.1.

The poset {2, 3, 4, 6} under divisibility is represented by {see Fig. 6.1}

If $X = \{1, 2, 3\}$ then.

$$P(X) = \{\phi, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$$

under $\subseteq$ relation is represented by (Fig. 6.2).

A chain would always have the shape as show in Fig. 6.3 which also justifies the name chain for this poset.

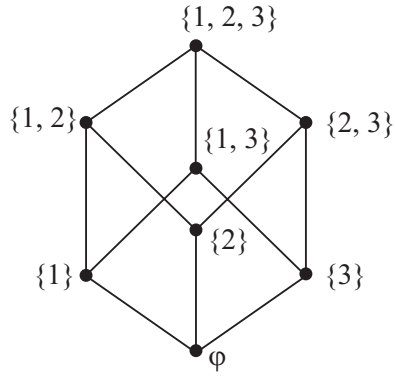


Fig. 6.2

Fig. 6.3 (a chain)

Definition: Let P be a poset. If there exists an element $a \in P$ such that $x \leq a$ for all $x \in P$ then a is called greatest or unity elements of P.

Also an element $b \in P$ will be called *least or zero element* of P if $b \leq x$ $\forall\ x \in P$. It is denoted by o. Least element (if it exists) will be unique. The following example gives us some more information about greatest and least elements, which are also called the *universal bounds of the poset*. In fact, if a poset P has least and greatest elements, we call it a bounded poset. Indeed $o \leq x \leq u\ \forall\ x \in P$.

An element a in a poset P is called maximal element of P if $a < x$ for no $x \in P$.

6.2.3 Isomorphism

Let (P, R) and (Q, R') *be two posets*. A *one-one onto map* $f : P \to Q$ *is called an isomorphism if* $x\,R\,y \Leftrightarrow f(x)\,R'\,f(y), \forall\, x, y \in P$.

As *agreed* earlier and if there is no scope of confusion we use the same symbol $\leq$ for both the relations R and R' and thus our definition translates to.

A one-one onto map $f : P \to Q$ is called an isomorphism if $x \leq y \Leftrightarrow f(x) \leq f(y)$. We write, in that case, $P \cong Q$. It is easy to show that the relation of isomorphism is an equivalence relation.

In fact a map $f : P \to Q$ is called *isotone* or *order preserving* if $x \leq y \Rightarrow f(x) \leq f(y)$.

SOLVED EXAMPLES

Example 1

Show that in a poset $a < a$ *for no* a *and* $a < b, b < c \Rightarrow a < c$.

Solution

Let there exists some element a in poset P such that $a < a$.

Then by definition, $a \leq a$ and $a \neq a$

By anti-symmetry, $a \leq a, a \leq a \Rightarrow a = a$

Thus, we get a contradiction.

Again $a \leq b, b \leq c \Rightarrow a \leq b, a \neq b.$

$b \leq c, b \neq c$

So $a \leq b, b \leq c \Rightarrow a \leq c$

If $a = c$, then $c \leq b, b \leq c \Rightarrow b = c$, a contradiction.

Hence $a < c$. ■

Example 2

Let $X = \{1, 2, 3\}$ *then* $(P(X), \subseteq)$ *is a poset.*

Solution

Let $A = \{\phi, \{1, 2\}, \{2\}, \{3\}\}$

Then $(A, \subseteq)$ is a poset with ϕ as least element. A has no greatest element.

Let $B = \{\{1, 2\}, \{2\}, \{3\}, \{1, 2, 3\}\}$ then B has greatest element $\{1, 2, 3\}$ but no least element.

If $C = \{\phi, \{1\}, \{2\}, \{3\}\}$ Then C has both least and greatest element, namely ϕ and $\{1, 2\}$.

Again $D = \{\{1\}, \{2\}, \{1, 3\}\}$ has neither least nor greatest element.

■

Example 3

If S is non-empty finite subset of a poset P then S has maximal and minimal elements.

Solution

Let $x_1, x_2, ..., x_n$ be all the distinct elements of S is any random order.

If x_1 is maximal element, we are done.

If x_1 is not maximal then there exists some $x_i \in S$ such that $x_i < x_j$

If x_i is maximal, we are done. If not, there exists some $x_j \in S$ such that $x_i < x_j$

Continuing likewise, we reach a stage where some elements will be maximal.

Similarly, we can show that S has maximal element. ■

Example 4

A mapping $f : P \to Q$ is an isomorphism if and only if f is isotone and has an isotone inverse.

Solution

Let $f : P \to Q$ be an isomorphism.

Then f being 1 – 1, onto, f^{-1} exists is 1 – 1, onto.

Again by definition of isomorphism, f will be isotone. We show $f^{-1}: Q \to P$ is also isotone.

Let $y_1, y_2 \in Q$ where $y_1 \le y_2$. Since f is onto, $\exists\, x_1, x_2 \in P$, s.t.,

$$f(x_1) = y_1, f(x_2) = y_2$$

$\Leftrightarrow$ $x_1 = f^{-1}(y_1), x_2 = f^{-1}(y_2)$

Now $y_1 \le y_2 \Rightarrow f(x_1) \le f(x_2)$

$\Rightarrow$ $x_1 \le x_2$ (Def. of isomorphism)

$\Rightarrow$ $f^{-1}(y_1) \le f^{-1}(y_2)$

$\Rightarrow f^{-1}$ is isotone

Conversely, let f be isotone such that f^{-1} is also isotone. Since f^{-1} exists, f is 1-1, onto.

Again, as f is isotone $x_1 \le x_2 \Rightarrow f(x_1) \le f(x_2), x_1, x_2 \in P$

Also f^{-1} is isotone implies

$$f(x_1) \le f(x_2) \Rightarrow f^{-1}(f(x_1)) \le f^{-1}(f(x_2))$$

$\Rightarrow$ $x_1 \le x_2$

thus $x_1 \le x_2 \Leftrightarrow f(x_1) \le f(x_2)$

Hence f is an isomorphism. ■

6.2.4 Duality

Let ρ be a relation defined on a set X. Then converse of ρ denoted by $\overline{\rho}$ is defined by $a\ \overline{\rho}\ b \Leftrightarrow b\ \rho a$, $a, b \in X$ *i.e.,* dual of a poset is a poset.

Definition: If (X, ρ) be a poset then the poset $(\overline{X}, \overline{\rho})$, where $\overline{X} = X$ and $\overline{\rho}$ is converse of ρ is called dual of X.

Self Dual: If a poset X is isomorphic to its *dual* $\overline{X}$, then X is called *self dual*.

Note: Two posets are said to be dually isomorphic if one is isomorphic to the dual of the other.

SOLVED EXAMPLES

Example 1

If a set X forms a poset under a relation ρ *then prove that X forms a poset under* $\overline{\rho}$*, the converse of* ρ.

Solution

$a \,\overline{\rho}\, a$ as $a \,\rho\, a \;\forall\; a \in X$ shows $\overline{\rho}$ is reflexive.

Let $a \,\overline{\rho}\, b$ and $b \,\overline{\rho}\, a$, then $b \,\rho\, a$ and $a \,\rho\, b$ *i.e.,* $a \,\rho\, b$ and $b \,\rho\, a \Rightarrow a = b$.

Thus $\overline{\rho}$ is anti-symmetric

Let $a \,\overline{\rho}\, b$, $b \,\overline{\rho}\, c$ then $b \,\rho\, a$, $c \,\rho\, b$

or $c \,\rho\, b$, $b \,\rho\, a$

$\Rightarrow c \,\rho\, a \Rightarrow a \,\overline{\rho}\, c$

or that $\overline{\rho}$ is transitive and hence is a partial ordering. ■

Example 2

If (X, ρ) be a poset, then prove that $X \cong \overline{\overline{X}}$ *where* $\overline{\overline{X}}$ *is dual of* $\overline{X}$.

Solution

Define $f : X \to \overline{\overline{X}}$ such that $f(x) = x, \forall\, x \in X$.

f is then clearly a well defined one-one onto map again $x \,\rho\, y \Leftrightarrow y \,\overline{\rho}\, x \Leftrightarrow x \,\overline{\overline{\rho}}\, y \Leftrightarrow f(x) \,\overline{\overline{\rho}}\, f(y)$ shows f is an isomorphic ($\overline{\overline{\rho}}$ being converse of $\overline{\overline{\rho}}$).

■

Example 3

The poset X = {2, 3, 4, 6} of non-trivial factors of 12 under divisibility is self dual prove it.

Solution

Define $f : X \to \overline{X}$ s.t.,

$$f(x) = \frac{12}{x}$$

then f is clearly 2 1 – 1 onto mapping.

Also $a \le b$ in $X \Leftrightarrow a \mid b \Leftrightarrow b = ka$

$$\Leftrightarrow \quad \frac{12}{b} = \frac{12}{ka} \quad \Leftrightarrow \quad k \cdot \frac{12}{b} = \frac{12}{a}$$

$$\Leftrightarrow \quad \frac{12}{a} \text{ is a multiple of } \frac{12}{b}$$

$$\Leftrightarrow \quad f(a) \le f(b) \text{ in } \overline{X}$$

Notice converse of the relation 'divides' will be 'is a multiple of'. ■

Example 4

The two chains

$$S = \left\{0, \ldots \frac{1}{n}, \ldots, \frac{1}{3}, \frac{1}{2}, 1\right\}, \leq$$

$$T = \left\{0, \frac{1}{2}, \frac{2}{3}, \ldots \frac{n}{n+1}, \ldots 1\right\}, \leq$$

are dually isomorphic. Prove it.

Solution

Consider the map $f : \overline{S} \to T$. s.t.,

$$f(0) = 1$$

$$f\left(\frac{1}{n+1}\right) = \frac{n}{n+1},\ n = 0, 1, 2, 3, \ldots$$

Clearly f is a well defined 1-1 onto map.

Again $\quad \frac{1}{n+1} \geq \frac{1}{m+1}$ in $\overline{S}$

$\Leftrightarrow \quad n + 1 \leq m + 1$

$\Leftrightarrow \quad n \leq m \Leftrightarrow n + nm \leq m + nm$

$\Leftrightarrow \quad \frac{n}{n+1} \leq \frac{m}{m+1} \Leftrightarrow f\left(\frac{1}{n+1}\right) \leq f\left(\frac{1}{m+1}\right)$ in T. ■

Remark: We have defined that a poset A is dually isomorphic to a poset B if $A \cong \overline{B}$. Now if $B = A$ then A is dually isomorphic to $A \cong \overline{A}$ *i.e.,* when A is self dual. So a poset is dually isomorphic **to** itself iff A is self dual. In fact a 1-1 onto map $f : A \to B$ is called a dual isomorphism if $x\, R\, y \Leftrightarrow f(y)\, R'\, f(x)$ where R and R' are the relation of A and B.

Thus if $B = A$, then $f : A \to A$ is a dual isomorphism if it is 1-1 onto and $x\, R\, y \Leftrightarrow f(y)\, R\, f(x)$ *i.e.,* $x\, R\ \ y \Leftrightarrow f(x)\, \overline{R}\, f(y)$ or that f is an isomorphism from A to $\overline{A}$.

6.2.5 Product of two Posets

Let A and B be two posets, then, we can show that

$$A \times B = \{(a, b) : a \in A, b \in B\}$$

forms a poset under the relation defined by.

$$(a_1, b_1) \le (a_2, b_2) \Leftrightarrow a_1 \le a_2 \text{ in } A$$
$$b_1 \le b_2 \text{ in } B$$

It is clear that the three relations occurring above are different, being the respective relations in $A \times B$, A and B.

***Reflexivity*:** $(a, b) \le (a, b)\ \forall\ (a, b) \in A \times B$

as $a \le a$ in A and $b \le b$ in $B\ \forall\ a \in A,\ b \in B$

***Anti symmetry*:** Let $(a_1, b_1) \le (a_2, b_2)$ and $(a_2, b_2) \le (a_1, b_1)$

Then $a_1 \le a_2, b_1 \le b_2$, and $a_2 \le a_1, b_2 \le b_1$

$\Rightarrow$ $a_1 = a_2, b_1 = b_2 \quad \Rightarrow \quad (a_1, b_1) = (a_2, b_2)$

***Transitivity*:** Let $(a_1, b_1) \le (a_2, b_2)$ and $(a_2, b_2) \le (a_3, b_3)$

Then $a_1 \le a_2, b_1 \le b_2$ and $a_2 \le a_3, b_2 \le b_3$

$\Rightarrow$ $a_1 \le a_3$ and $b_1 \le b_3 \quad \Rightarrow \quad (a_1, b_1) \le (a_3, b_3)$.

We thus conclude that *product of two posets is a poset*. It is also called the direct or *cardinal product* of posets. The definition can be extended to product of more than two posets in a similar way.

Again, if we denote by $o(A)$ = number of elements in A then $o(A \times B) = o(A) \times o(B)$ follows by definition. (We read $o(A)$ as order A)

For example: The cardinal product of two self dual posets self dual.

Let A and B be the given self dual posets.

Let $f: A \to \overline{A}$ and $g: B \to \overline{B}$ be the isomorphisms.

Define $h: A \times B \to \overline{A \times B}$, s.t.,

$$h((a, b)) = \{(f(a), g(b))\}$$

then h is well defined, one-one map as

$$(a_1, b_1) = (a_2, b_2)$$
$\Leftrightarrow$ $a_1 = a_2, b_1 = b_2$

$\Leftrightarrow$ $f(a_1) = f(a_2), g(b_1) = g(b_2)$

$\Leftrightarrow$ $(f(a_1), g(b_1) = f(a_2), g(b_2))$

$\Leftrightarrow$ $h((a_1, b_1)) = h((a_2, b_2))$

Again for any $(x, y) \in \overline{A \times B}$, since

$$h(f^{-1}(x), g^{-1}(y)) = (ff^{-1}(x), gg^{-1}(y)) = (x, y)$$

and f^{-1}, g^{-1} exists as f, g are 1-1 onto, we find h is onto.

Again $(a_1, b_1) \le (a_2, b_2)$

$$\Leftrightarrow \quad a_1 \le a_2, b_1 \le b_2$$
$$\Leftrightarrow \quad f(a_1) \ge f(a_2), g(b_1) \ge g(b_2)$$
$$\Leftrightarrow \quad (f(a_1)\, g(b_1)) \ge (f(a_2), g(b_2))$$
$$\Leftrightarrow \quad h((a_1, b_1)) \ge h((a_2, b_2))$$

Thus h is the required isomorphism and $A \times B \cong \overline{A \times B}$.

6.3 Lattices as Posets

A poset $(\mathbf{L}, \le)$ is said to be a lattice if for every $a, b \in \mathbf{L}$, *sup* $\{a, b\}$ and *inf* $\{a, b\}$ exists in **L**.

In that case, we may write

$$sup\ \{a, b\} = a \vee b \qquad \text{(read } a \text{ join } b\text{)}$$
$$inf\ \{a, b\} = a \wedge b \qquad \text{(read } a \text{ meet } b\text{)}$$

Some other notations like $a + b$ and $a \cdot b$ or $a \cup b$ and $a \cap b$ are also used for sup $\{a, b\}$ and inf $\{a, b\}$.

The **greatest lower bound** (*glb*) of a subset $\{a, b\} \subseteq \mathbf{L}$ will be denoted by $a * b$ and the **least upper bound** (*lub*) by $a \oplus b$.

It follows from above, both * and $\oplus$ are binary operation on L because of the uniqueness of the *lub* and *glb* of any subset of a partially ordered set.

A totally ordered set is trivially a lattice, but not all partially ordered sets are lattices, as can be seen from the Hasse diagram of some of the partially ordered sets as shown in Fig. 6.3 and 6.4.

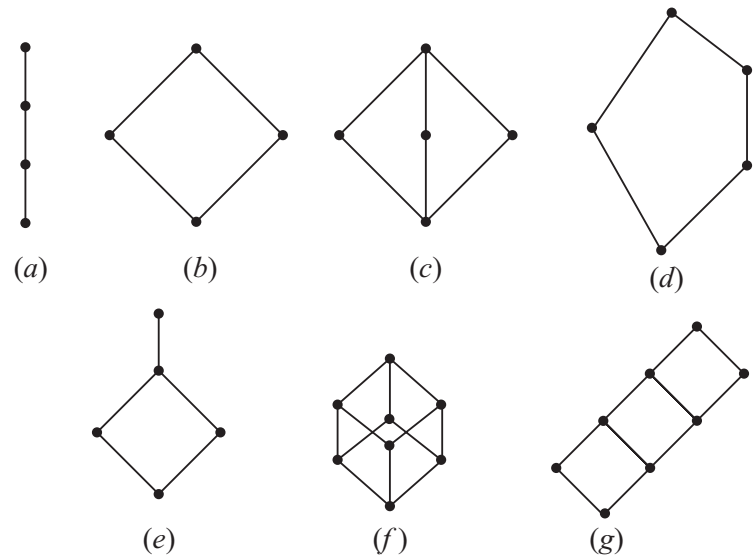


Fig. 6.3. Posets which are lattices.

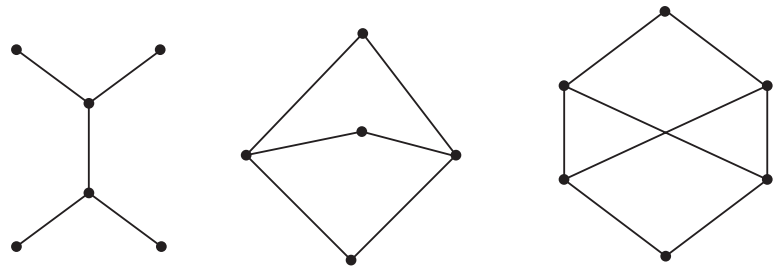


Fig. 6.4. Posets which are not Lattices.

6.3.1 Some Properties of Lattices

In any lattice $(\mathbf{L}, \leq)$, the following are true for each $a, b, c \in \mathbf{L}$:

$(\mathbf{L}_1)$ $a \wedge a = a$; $(\mathbf{L}_1)'$ $a \vee a = a$ (Idempotence)

$(\mathbf{L}_2)$ $a \wedge b = b \wedge a$; $(\mathbf{L}_2)'$ $(a \vee b) = (b \vee a)$ (Commulativity)

$(\mathbf{L}_3)$ $(a \wedge b) \wedge c = a \wedge (b \wedge c)$; $(\mathbf{L}_3)'$ $(a \vee b)\, c = a \vee (b \vee c)$; (Associativity)

$(\mathbf{L}_4)$ $a \wedge (a \vee b) = a$; $(\mathbf{L}_4)'$ $a \vee (a \wedge b) = a$ (Absorption)

Isotonicity: For each $a, b, c \in \mathbf{L}$.

we have $a \leq b \Rightarrow (a \wedge b) \leq (b \wedge c)$

and $(a \vee c) \leq (b \vee c)$

i.e., join and meet taken with a fixed element, preserve the inequality.

i.e., If $a \leq b$ then $a \leq b \leq b \vee c$, $c \leq b \vee c$ given that $b \vee c$ is an upper bound of $\{a, b\}$.

$\therefore$ $a \vee c \leq b \vee c$, $a \vee e$ being the least among the upper bounds of $\{a, c\}$.

Similarly, $a \wedge c \leq a \leq b$.

$\therefore$ $a \wedge c$ is a lower bound of $\{b, c\}$ and so,

$a \wedge c \leq b \wedge c$ *glb* of $\{a, b\}$.

Stability: For any $a, b, c \in \mathbf{L}$ we have

(*i*) $a \leq b,\ a \leq c \Rightarrow a \leq (b \vee c),\ a \leq (b \wedge c)$

(*ii*) $a \geq b,\ a \geq c \Rightarrow a \geq (b \wedge c),\ a \geq (b \vee c)$

Since (*ii*) is of course the dual of (*i*)

To see (*i*), observe that if $a \leq b,\ a \leq c$ holds.

Then $a \leq b \leq b \vee c$ always, and $a \leq b,\ a \leq c \Rightarrow a$ is lower bound of $\{b, c\}$, therefore $a \leq$ *greatest lower bound* $\{b, c\} = b \wedge c$.

Distributive inequalities: For each $a, b, c \in \mathbf{L}$, the following distributive inequalities hold in any lattice.

(i) $\quad a \wedge (b \vee c) \geq (a \wedge b) \vee (a \wedge c)$

(ii) $\quad a \vee (b \wedge c) \leq (a \vee b) \wedge (a \vee c)$

Modular inequality: If $(\mathbf{L}, \leq)$ be any lattice, then for any $a, b, c \in \mathbf{L}$, we have

$$a \leq c \Leftrightarrow a \vee (b \vee c) \leq (a \vee b) \vee c.$$

6.3.2 Lattices as Algebraic Systems

We define on n-ary operation on a set X as function O from $X^n = X \times X \times X \ldots \times X$ (Cartesian product n copies of X) the case $n = 2$, the most important in which it is called a binary operation. Here by X^0 we mean a singleton set.

An algebraic system is a set X together with collection of n-ary operations on X satisfying certain conditions.

We had introduced lattices as partially ordered sets in which for each $a, b \in X$, sup $\{a, b\} = a \vee b$ and inf $\{a, b\} = a \wedge b$ exists in the set. This gives two binary operations on the underlying set of a lattice given by

$V : \mathbf{L} \times \mathbf{L} \Rightarrow$ and $\wedge : \mathbf{L} \times \mathbf{L} \rightarrow \mathbf{L}$

given by $\vee ((a, b)) = a \vee b$ and $\wedge (a, b) = a \wedge b$.

Also we have seen that $\wedge$ and $\vee$ satisfy certain properties like idempotence, absorption, commutative and associativity and other results like modular and distributive inequalities hold involving the operations $\vee$ and $\wedge$, as well as the order relation $\leq$.

Lattice Homomorphism: If $(\mathbf{L}, \wedge, \vee)$ and $(\mathbf{L}', *, \oplus)$ are two lattices, then a lattice homomorphism is a function $f : \mathbf{L} \Rightarrow \mathbf{L}'$ such that

$$f(a \wedge b) = f(a) * f(b)$$

and $f(a \vee b) = f(a) \oplus (b)$ hold for all a and b, in $\mathbf{L}$. On the other hand an order preserving map g between two posets $(\mathbf{L}, \leq)$ and $(\mathbf{L}, \leq)$ is one for which $a \leq b \Rightarrow g(a) \leq g(b)$.

Note: Bijective lattice homomorphisms are called lattice isomorphism. A bijective map f such that both the maps f and $f - 1$ are order preserving, is called an **order isomorphism**.

SOLVED EXAMPLES

Example 1

The set L = {1, 2, 3, 4, 5, 6, 12} of factors of 12 under divisibility forms a lattice. Prove by Hasse diagram.

Solution

When the partial order relation is divisibility then

$$a \wedge b = g.c.d.\ (a, b)$$

$$a \vee b = 1.c.m.\ (a, b)$$

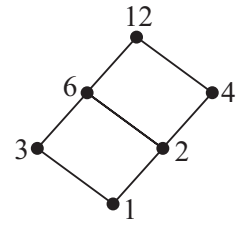


Fig. 6.5.

$a \wedge b$ by def. is *inf* $\{a, b\}$ and if *inf* $\{a, b\} = x$ then we should have $x \le a$, $x \le b$ and if $y \le a$, $y \le b$ then $y \le x$ which translates to saying $x \mid a$, $x \mid b$ and if $y \mid a, y \mid b$ then $y \mid x$.

Now by def., *g.c.d.* $(a, b) = c$ means $c \mid a$, $c \mid b$ and if $d \mid a$, $d \mid b$ then $d \mid c$.

In other words we find, *g.c.d.* (a, b) is precisely $a \wedge b$. Similarly *l.c.m.* $(a, b) = a \vee b$. ■

Example 2

Let S be any set and L be a lattice. Let $\top$ *= set of all functions from* $S \to L$. *Define a relation* $<$ *on* $\top$ *by*

$$f \le g \Leftrightarrow f(x)\ g(x)\ \forall\ x \in S, f, g \in \top \text{ prove that.}$$

($\top$*,* $\le$*) forms a poset. Show that (*$\top$*,* $\le$*) is a lattice*

Solution

Let $f, g \in \top$ be any two members, then f, g are functions $S \to L$.

Let $\theta : S \to L$, s.t.,

$$\theta(x) = f(x) \vee g(x)\ \forall\ x \in S$$

$$x \in S \Rightarrow f(x), g(x) \in L \Rightarrow f(x) \vee g(x) \in L\ (L \text{ being a lattice})$$

We show $\theta = \text{Sup}\ \{f, g\}$.

Now $\quad \theta(x) = f(x) \vee g(x)$ in $L \Rightarrow f(x) \le \theta(x)$

$$g(x) \le \theta(x)$$

$\Rightarrow \quad f \le \theta, g \le \theta$

$\Rightarrow \theta$ is an upper bound of $\{f, g\}$ in $\top$

Let ϕ be any upper bound of $\{f, g\}$ in $\top$, then

$$f \le \phi,\ g \le \phi, \Rightarrow f(x) \le \phi(x)$$

$$g(x) \le \phi(x)$$

$$\Rightarrow \quad f(x) \vee g(x) \le \phi(x)$$

$$\Rightarrow \quad \theta(x) \le \phi(x) \Rightarrow \theta \le \phi$$

Hence $\theta = \text{Sup}\{f, g\}$.

Similarly, we can show that $\psi : S \to L$, s.t.,

$$\psi(x) = f(x) \wedge g(x),\ \forall\ x \in S$$

will be Inf $\{f, g\}$

Hence $\top$ is a lattice. ■

Example 3

Draw the diagram of the lattice of factors of 20, under divisibility and show that it is same as that of the product of two chains with three and two elements.

Solution

We write 20 as product of prime powers,

we find, $20 = 2^2 \times 5^1$

Number of factors is $(2 + 1)(1 + 1) = 6$,

and factors are $2^\alpha \times 5^\beta$ where $\alpha = 0, 1, 2$, $\beta = 0, 1$

i.e., the six factors are, $2^0 \times 5^0, 2^1 \times 5^0, 2^2 \times 5^0$

$$2^0 \times 5^1, 2^1 \times 5^1, 2^2 \times 5^1$$

i.e., 1, 2, 4, 5, 10, 20.

the diagram representing this lattice will be,

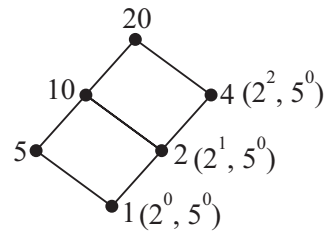


Fig. 6.6.

If we consider the two chains

$C_1 = \{0, 1, 2\}$ *i.e.* $(\alpha = 0, 1, 2)$

$C_2 = \{0, 1\}$ *i.e.* $(\beta = 0, 1)$

Then $C_1 \times C_2 = \{(0, 0), (1, 0), (2, 0), (0, 1), (1, 1), (2, 1)\}$ and it would also have the same diagram as $C_1 \times C_2$ as shown in Fig. 6.7. ■

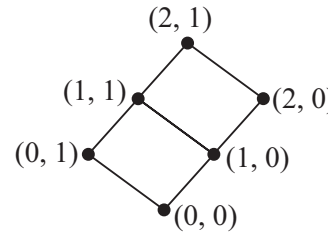


Fig. 6.7.

Example 4

Let $\mathbf{L}$ *= (P ({a, b, c}),* $\wedge$*,* $\cup$*) and* $\mathbf{L}$ *= (D (30), g.c.d., l.c.m).*

Define f : $\mathbf{L} \to \mathbf{L}'$ *by.*

$$f(\leq) = a\, f(\{a\}) = 2, f(\{b\} = 3,$$

$$f(\{c\}) = 5$$

$$f(\{a, b\} = 6, f(\{b, c\}) = 15,$$

$$f(\{a, c\} = 10, \text{ and } f(\{a, b, c\}) = 30.$$

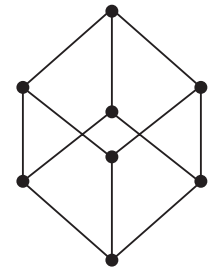


Fig. 6.8.

Verify that if lattice isomorphisms. Both have identied Hasse diagram.

Solution

It is easy to see that every lattice homomorphism is order preserving. In fact preservation of any one of the lattice operations imply that it is order preserving thus if ($\mathbf{L}$, $\wedge$, $\vee$) and ($\mathbf{L}'$,* $\oplus$) are any two lattices with the corresponding order relations denoted by $\leq$ and $\leq'$, then if $f(a \wedge b) = f(a) * f(b)$ $\forall$ a, b $\mathbf{L}$, then whenever $a \leq b$ holds, we have $a \wedge b = a$ and so

$$f(a) = f(a \wedge b) = f(a) * f(b) \;\therefore f(a) \leq f(b).$$

But order preservation does not guarantee that it is a lattice homomorphism, even if it is an injective map. Let $\mathbf{L}$ = set of all subspaces of a vector space V and $\mathbf{L}' = P(V)$. We have the lattice operations as + (Linear sum of subspace) and $\cap$ in $\mathbf{L}$ and $\cup$, $\cap$, $\mathbf{L}'$. Define f: $\mathbf{L} \Rightarrow \mathbf{L}'$ by $f(H) = H$. (Considered as a set) $\forall$ subspace H of V.

Then $f(H_1 \vee H_2) = f(H_1 + H_2) = H_1 + H_2$ whereas $f(H_1) \cup f(H_2) = H_1 \cup H_2$ and the two are unequal unless one of them is contained in the order. So f is not a lattice homomorphism, although it is order preserving, for the order is set inclusion in both cases and $f(H) = H$.

Theorem 1: In any lattice the distributive inequalities

$$a \wedge (b \vee c) \geq (a \wedge b) \vee (a \wedge c)$$

$$a \vee (b \wedge c) \leq (a \vee b) \wedge (a \vee c)$$

holds for any $a, b, c \in \mathbf{L}$

Proof: $a \wedge b \leq a, a \wedge b \leq b \leq b \vee c,$

$\Rightarrow a \wedge b$ is lower bound of $\{a, b \vee c\}$.

$$= a \wedge b \leq a \wedge (b \vee c) \qquad \ldots(i)$$

Again $a \wedge c \leq a$

$a \wedge c \leq c \leq b \vee c$

$\Rightarrow$ $a \wedge c \leq a \wedge (b \vee c)$ …(*ii*)

from (*i*) and (*ii*), $a \wedge (b \vee c)$ is upper bound of $\{a \wedge b, a \wedge c\}$

$\Rightarrow$ $(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee c)$

Similarly, other inequality can also be proved.

Theorem 2: In any lattice **L**,

$$(a \wedge b) \vee (b \wedge c) \vee (c \wedge a) \leq (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$$

for all $a, b, c \in \mathbf{L}$.

Proof: Since $a \wedge b \leq a \vee b$.

$a \wedge b \leq b \leq b \vee c$

$a \wedge b \leq a \leq c \vee a$

we find, $(a \wedge b) \leq (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$

similarly, $(b \wedge c) \leq (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$

and $(c \wedge a) \leq (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$

Hence $(a \wedge b) \vee (b \wedge c) \vee (c \wedge a) \leq (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$.

Note: (*i*) A poset $(P, \leq)$ is called a meet semi lattice if for all $a, b \in P$, inf $\{a, b\}$ exists.

(*ii*) A non empty set P together with a binary composition $\wedge$ is called a meet-semi lattice if for all $a, b, c \in P$.

s.t., $a \wedge a = a, a \wedge b = b \wedge a, a \wedge (b \wedge c) = (a \wedge b) \wedge c$.

Theorem 3: Dual of a lattice is a lattice.

Proof: Let $(\mathbf{L}, R)$ be a lattice and let $(\overline{\mathbf{L}}, \overline{R})$ be its dual. Then $\overline{\mathbf{L}} = \mathbf{L}$ and $\overline{R}$ is converse of R. We have already proved that dual of a poset is a poset.

Let $x, y \in \overline{\mathbf{L}}$ be any elements, then $x, y \in \mathbf{L}$ and as **L** as is a lattice, Sup $\{x, y\}$ exists in **L**. Let it be $x \vee y$

Then $x \, R \, (x \vee y)$

$y \, R \, (x \vee y)$

$\Rightarrow$ $(x \vee y) \, \overline{R} \, x$

$(x \vee y) \, \overline{R} \, y$

$\Rightarrow$ $x \vee y$ is a lower bound of $\{x, y\}$ in $\overline{\mathbf{L}}$.

If z is any lower bound of $\{x, y\}$ in $\overline{\mathbf{L}}$ then

$z \, \overline{R} \, x, z \, \overline{R} \, y$

$\Rightarrow$ $x\ R\ z,\ y\ R\ z$

$\Rightarrow$ z is an upper bound of $\{x, y\}$ in $\mathbf{L}$

$\Rightarrow$ $(x \vee y)\ R\ z$ as $x \vee y = \text{Sup}\ \{x, y\}$ in $\mathbf{L}$

$\Rightarrow$ $z\ \overline{R}\ (x \vee y)$

or that $x \vee y$ is greatest lower bound of $\{x, y\}$ in $\overline{\mathbf{L}}$. Similarly we can show $x \wedge y$ will be Sup $\{x, y\}$ in $\overline{\mathbf{L}}$.

Hence $\overline{\mathbf{L}}$ is a lattice. ■

Theorem 4: Product of two lattices is a lattice.

Proof: Let A and B be two lattices then we have already proved that $A \times B = \{(a, b) \mid \{a \in A, b \in B\}$ is a poset under the relation $\leq$ defined by

$$(a_1, b_1) \leq (a_2, b_2) \Leftrightarrow a_1 \leq a_2 \text{ in } A$$
$$b_1 \leq b_2 \text{ in } B.$$

We show $A \times B$ forms a lattice.

Let $(a_1, b_1), (a_2, b_2) \in A \times B$ be any elements.

Then $a_1, a_2 \in A$ and $b_1, b_2 \in B$.

Since A and B are lattices, $\{a_1, a_2\}$ and $\{b_1, b_2\}$ have Sup and Inf in A and B respectively.

Let $a_1 \wedge a_2 = \text{Inf}\ \{a_1, a_2\}$, $b_1 \wedge b_2 = \text{Inf}\ \{b_1, b_2\}$

then $a_1 \wedge a_2 \leq a_1,\ a_1 \wedge a_2 \leq a_2$

$b_1 \wedge b_2 \leq b_1,\ b_1 \wedge b_2 \leq b_2$

$\Rightarrow$ $(a_1 \wedge a_2, b_1 \wedge b_2) \leq (a_1, b_1)$

$(a_1 \wedge a_2, b_1 \wedge b_2) \leq (a_2, b_2)$

$\Rightarrow (a_1 \wedge a_2, b_1 \wedge b_2)$ is a lower bound of $\{(a_1, b_1), (a_2, b_2)\}$

Suppose (c, d) is any lower bound of $\{(a_1, b_1), (a_2, b_2)\}$

Then $(c, d) \leq (a_1, b_1)$

$(c, d) \leq (a_2, b_2)$

$\Rightarrow$ $c \leq a_1, c \leq a_2, d \leq b_1, d \leq b_2$

$\Rightarrow c$ is a lower bound of $\{a_1, a_2\}$ in A.

d is a lower bound of $\{b_1, b_2\}$ in B.

$\Rightarrow$ $c \leq a_1 \wedge a_2 = \text{Inf}\ \{a_1, a_2\}$

$d \leq b_1 \wedge b_2 = \text{Inf}\ \{b_1, b_2\}$

$\Rightarrow$ $(c, d) \leq (a_1 \wedge a_2, b_1 \wedge b_2)$

or that $(a_1 \wedge a_2, b_1 \wedge b_2)$ is *g.l.b.* $\{(a_1, b_1), (a_2, b_2)\}$.

Similarly (by duality) we can say that

$$(a_1 \vee a_2, b_1 \vee b_2) \text{ is } l.u.b \{(a_1, b_1), (a_2, b_2)\}$$

Hence $A \times B$ is a lattice.

Also $(a_1, b_1) \wedge (a_2, b_2) = (a_1 \wedge a_2, b_1 \wedge b_2)$

$\forall\ (a_1, b_1) \vee (a_2, b_2) = (a_1 \vee a_2, b_1 \vee b_2)$.

Theorem 5: A finite lattice has least and greatest elements.

Proof: Let $L = \{x_1, x_2, ..., x_n\}$ be the given finite lattice suppose $x_1, x_2, ..., x_n$ the elements, written in random order,

put $a_0 = x_1, a_1 = a_0 \wedge x_1, a_2 = a_1 \wedge x_2, ..., a_n = a_{n-1} \wedge x_n$

Than $a_n \leq a_{n-1} \leq .... \leq a_0$ and $a_i \leq x_i\ \forall\ i = 1, 2, ..., n$.

i.e., a_n is least element of **L**.

Similarly, we can find greatest element of L.

6.3.3 Complete Lattice

A lattice **L** is called a complete lattice if every nonempty subset of **L** has its *sup* and *inf* in **L**.

The lattice $(Z, \leq)$ of integers is not complete as the subset $k = \{x \in Z : x > 0\}$ does not have an upper bound and therefore a *sub* in Z.

The real interval [0, 1] with usual $\leq$ forms a complete lattice.

Note: (*i*) If **L** is a complete lattice then it must have the least and the greatest elements, since **L** being its own subset must have *sup* and inf.

(*ii*) Dual of a complete lattice is complete.

6.3.4 Bounded Lattice

A lattice (**L**,) is said to be bounded if $\exists\ 0, 1 \in \mathbf{L}$ such that $0 \leq a \leq 1$ for every $a \in L$.

Therefore we have

1. Every finite lattice **L** is bounded. Here $0 = Va_1$ and $1 = Va_1$, where $\mathbf{L} = \{a_1 ..., a_n\}$.
2. $(P(s), \cap, \cup)$ is bounded with $0 = \phi$ and $1 = s$.
3. $(IN,), (Z, \leq) (R, \leq), (\overline{R}, \leq)$ are not bounded.
4. (*IN, divides*) is not bounded. Here 1 is the lower bound for the lattice but there is no upper bound.
5. Divisor lattices are bounded. In $D(n)$. 1 is the lower bound and n is the upper bound.

6.3.5 Sublattices

Let $(\mathbf{L}, \wedge \vee)$ be a lattice and $\mathbf{L}' \subseteq \mathbf{L}$ be a subset, then $(\mathbf{L}', \wedge, \vee)$ is a sublattice of $(\mathbf{L}, \wedge, \vee)$ if $\mathbf{L}'$ is closed under $\wedge$ and $\vee$ *i.e.,* $a \wedge b, a \vee b \in \mathbf{L}'$ $\forall$ $a, b \in \mathbf{L}'$.

Thus $(\mathbf{L}', \wedge, \vee)$ is a lattice in its own right where $a \wedge b$ and $A' \vee b$ in $\mathbf{L}'$ have been taken as the corresponding meet and join of a and b considered as elements of $\mathbf{L}$.

Note: (*i*) Every nonempty subset of a chain is a sublattice

(*ii*) Union of two sublattices may not be a sublattice.

(*iii*) A lattice is a chain iff every nonempty subset of it is a sublattice.

(*iv*) A sublattice S of a lattice $\mathbf{L}$ is called a *convex sublattice* if for all $a, b \in S$, $[a \wedge b, a \vee b] \subseteq S$.

6.3.6 Ideals of Lattices

A non empty subset I of a lattice $\mathbf{L}$ is said to be an ideal of $\mathbf{L}$ if

(*a*) $\quad a, b \in I \Rightarrow a \vee b \in I$

(*b*) $\quad a \in I, l \in \mathbf{L} \Rightarrow a \wedge l \in I.$

If L is bounded then $\{0\}$ is always an ideal of L and known as **zero ideal**.

Note: (*i*) An ideal is a sublattice but converse is not true.

(*ii*) Intersection of two ideals is an ideal.

(*iii*) Union of two ideals is an ideal iff one of them is contained in the other.

(*iv*) The set of all ideals of a lattice $\mathbf{L}$ forms an ideal under $\subseteq$ relation.

Dual Ideal: A non empty subset I of a lattice $\mathbf{L}$ is called a dual ideal (a filter) of L if

(*a*) $\quad a, b \in I \Rightarrow a \wedge b \in I$

(*b*) $\quad a \in I, l \in \mathbf{L} \Rightarrow a \vee l \in I$

Note: (*i*) A dual ideal is a convex sublattice.

(*ii*) Intersection of dual ideals is a dual ideal.

(*iii*) Union of two dual ideals in a dual ideal iff one of them is contained in the other.

For example: The Set **L** = {1, 2, 4, 5, 10, 20 , 25, 50, 100} of factors of 100 forms a lattice under divisibility. It contains 9 ideals {1}, {1, 2}, {1, 2, 4}, {1, 5}, {1, 2, 5, 10}, {1, 2, 4, 5, 10, 20}, {1, 5, 25}, {1, 2, 5, 10, 25, 50}, {1, 2, 4, 5, 10, 20, 25, 50, 100}

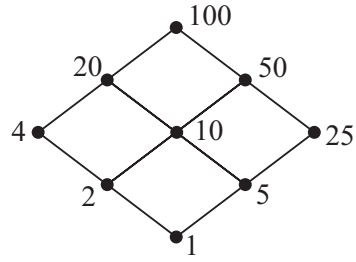


Fig. 6.9.

6.4 Modular and Distributive Lattices

In any lattice the distributive inequalities

$$a \wedge (b \vee c) \geq (a \wedge b) \vee (a \wedge c)$$

$$a \vee (b \wedge c) \leq (a \vee b) \wedge (a \vee c)$$

hold for all *a, b* and *c*. In many important lattices, including Boolean Algebras, the above inequalities are actually equalities. Wherever this holds, the lattices are called distributive. We first observe that equality of The two sides of *al a, b, c* for either of the above distributive inequalities imply that of the other.

A lattice $(\mathbf{L}, \cap, \cup)$ is said to be distributive if the equivalent conditions of the theorem 6 below holds.

Theorem 6: In any lattice $(\mathbf{L}, \wedge, \vee)$ the following statements are equivalent.

(*i*) $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge b) \; \forall \; a, b, c \in \mathbf{L}$.

(*ii*) $a \vee (a \wedge b) = (a \vee b) \wedge (a \vee c) \wedge (a \vee c) \; \forall \; a, b, c \in \mathbf{L}$.

Proof: (*i*) $\Rightarrow$ (*ii*)

R.H.S. of (*ii*) = $[(a \vee b) \wedge a] \vee [(a \vee b) \wedge c]$ by (*i*)

$= a \vee [c \wedge (a \vee b)]$ (by commutativity and absorption)

$= a \vee [(c \wedge a) \vee (c \wedge b)]$ by (*i*)

$= [a \vee (c \wedge a)] \vee (c \wedge b)$ (by associativity)

$= a \vee (b \wedge c)$ (by absorption and commutativity)

= *L.H.S.* of (*ii*)

(*ii*) $\Rightarrow$ (*i*) may be proved in a dual manner.

For example: (*N, g.c.d, l.c.m*) is a distributive lattice.

To see this we have to show that

$$(a, [b, c]) = [(a, b), (a, c)].$$

i.e. $$g.c.d\ \{a, l.c.m.\{b, c\}\} = l.c.m.\{g.c.d.\{a, b\}.\ g.c.d, \{a, c\}\}$$

This readily follows from the decomposition of *a, b. c* as products of prime powers, By fundamental theorem of arithmetic, we write.

$$a = p_1^{a_n} \ldots p_n^{a_n}\ b = p_1^{b}{}_1 \ldots p_n^{b}{}_2$$

and $$c = p_1^{g_1} \ldots p_n^{g_n}$$

We may use the same set of primes by allowing the powers, α, β, γ, to be zero/we take $a, b, \gamma \geq 0$. For example

$$28 = 2^2 . 3^0 . 5^0 . 7^1$$

$$70 = 2^1 . 3^0 . 5^1 . 7^1$$

$$30 = 2^1 . 3^1 . 5^1 . 7^0$$

We know then that the *g.c.d* and *l.c.m* are expressed as the product of powers of the same set of primes but the powers now being the lesser or greater of the two powers.

Thus, $$[b,c] = \prod_{i=1}^{n} p_i \max(\beta_i, \gamma_i)$$

$$[a,c] = \prod_{i=1}^{n} p_i \max(\alpha_i, \beta_i)$$

$$[a,c] = \prod_{i=1}^{n} p_i \min(\alpha_i, \gamma_i)$$

$$\{a, [b,c]\} = \prod_{i=1}^{n} p_i \min(\alpha_i \max(\beta_i, \gamma_i))$$

While $$[(a, b), (a, c)] = \prod_{i=1}^{n} p_i \max[\min(\alpha_i, \beta_i), \min(\alpha_i, \gamma_i)]$$

Thus distributivity will hold if min. $(\alpha_i \max(\beta_i, \gamma_i)) = \max[\min(\alpha_i, \gamma_i), \min(\alpha_i, \gamma_i)]$ for all i. That this holds for any three numbers may be seen directly or is will also be observed in a later section on chains that every chain (*i.e.* totally ordered sets) is distributive lattice which is same as the above statements. In particular the divisor lattices $D(n)$ being sublattices of (N divides) are distributive.

Theorem 7: A sublattice of a modular lattice is modular.

Proof: Let S be a sublattice of a modular lattice $\mathbf{L}$. If $a, b, c \in S$ with $a \geq b$ then as $S \subseteq \mathbf{L}$, $a, b, c, \in \mathbf{L}$ and therefore

$$a \wedge (b \vee c) = b \vee (a \wedge c)$$

Since S is closed under $\wedge$ and $\vee$ this result holds in S and hence S is modular.

Theorem 8: A lattice of length two is modular.

Proof: If a lattice $\mathbf{L}$ has length 2, then $l\,[0, 4] = 2$ and then $l\,[a, b] \leq 2$ for any $a, b \in \mathbf{L}$, $a \leq b$.

Let $a, b, c \in \mathbf{L}$ be three elements such that, $a > b$ and c is not comparable with a or b. Then $b \wedge c < b < a < a \vee c$. Which shows $l\,[b \wedge c, a \vee c] \geq 3$. (not possible). Thus we cannot find any triplet a, b, c in $\mathbf{L}$ s.t. $a > b$ and c is not comparable with a or b.

Hence $\mathbf{L}$ is modular.

Theorem 9: Homomorphic image of modular lattice is modular.

Proof: Let $\theta : \mathbf{L} \to M$ be an onto homomorphism and suppose L is modular.

Let $x, y, z \in M$ be three elements with $x > y$.

since θ is onto homomorphism, there exists $a, b, c \in \mathbf{L}$ s.t.

$\theta(a) = x, \theta(b) = y, \theta(c) = z$ where $a > b$.

Now L is modular, $a, b, c \in \mathbf{L}$, $a > b$.

We get, $a \wedge (b \vee c) = b \vee (a \wedge c)$

Now
$$\begin{aligned} x \wedge (y \vee z) &= (\theta(a) \wedge \theta(b) \vee \theta(c)) \text{ (by definition)} \\ &= \theta(a) \wedge (\theta(b \vee c)) = \theta(a \wedge (b \vee c)) \\ &= \theta(b \vee (a \wedge c)) = \theta(b) \vee \theta(a \wedge c) \\ &= \theta(b) \vee [\theta(a) \wedge \theta(c)] = y \vee (x \wedge z) \end{aligned}$$

Hence M is modular.

Theorem 10: *Two lattices* $\mathbf{L}$ *and* M *are modular iff* $\mathbf{L} \times M$ *is modular.*

Proof: Let $(a_1, b_1), (a_2, b_2), (a_3, b_3) \in \mathbf{L} \times M$ be three elements with $(a_1, b_1) \geq (a_2, b_2)$.

Then
$$a_1, a_2, a_3 \in \mathbf{L}, a_1 \geq a_2$$
$$b_1, b_2, b_3 \in M, b_1 \geq b_2$$

and since L and M are modular, we get

$$a_1 \wedge (a_2 \vee a_3) = a_2 \vee (a_1 \wedge a_3)$$
$$b_1 \wedge (b_2 \vee b_3) = b_2 \vee (b_1 \wedge b_3)$$

Thus

$$\begin{aligned}(a_1, b_1) \wedge [(a_2, b_2) \vee (a_3, b_3)] &= (a_1, b_1) \wedge (a_2 \vee a_3, b_2 \vee b_3)\\ &= (a_1 \wedge (a_2 \vee a_3), b_1 \wedge (b_2 \vee b_3))\\ &= (a_2 \vee (a_1 \wedge a_3), b_2 \vee (b_1 \wedge b_3))\\ &= (a_2, b_2) \vee (a_1 \wedge a_3, b_1 \wedge b_3)\\ &= (a_2, b_2) \vee [(a_1, b_1) \wedge (a_3, b_3)]\end{aligned}$$

Hence $\mathbf{L} \times M$ is modular.

Conversely, Let $\mathbf{L} \times M$ be modular.

Let $a_1, a_2, a_3 \in \mathbf{L}, a_1 \geq a_2$

$b_1, b_2, b_3 \in M, b_1 \geq b_2$

Then $(a_1, b_1), (a_2, b_2), (a_3, b_3) \in \mathbf{L} \times M$ and $(a_1, b_1) \geq (a_2, b_2)$.

Since $\mathbf{L} \times M$ is modular, we find

$$(a_1, b_1) \wedge [(a_2, b_2) \vee (a_3, b_3)] = (a_2, b_2) \vee [(a_1, b_1) \wedge (a_3, b_3)]$$

or $$(a_1, b_1) \wedge (a_2 \vee a_3, b_2 \vee b_3)] = (a_2, b_2) \vee (a_1 \wedge a_3, b_1 \wedge b_3)$$

or $$(a_1 \wedge (a_2 \vee a_3), b_1 \wedge (b_2 \vee b_3)) = (a_2 \vee (a_1 \wedge a_3), b_2 \vee (b_1 \wedge b_3))$$

$$\Rightarrow \quad a_1 \wedge (a_2 \vee a_3) = a_2 \vee (a_1 \wedge a_3)$$

$$b_1 \wedge (b_2 \vee b_3) = b_2 \vee (b_1 \wedge b_3)$$

$\Rightarrow \mathbf{L}$ and M are modular.

Theorem 11: A lattice $\mathbf{L}$ is modular iff no interval $[x, y]$ of $\mathbf{L}$ contains an element which has two different comparable complements relative to $[x, y]$.

Proof: Suppose $\mathbf{L}$ is modular.

Suppose $[x, y]$ is an interval in $\mathbf{L}$ such that an element c in $[x, y]$ has two comparable complements a, b $(a \geq b)$ relative to $[x, y]$

Then $$c \wedge a = c \wedge b \; (= x)$$

$$c \vee a = c \vee b \; (= y)$$

Thus $$\begin{aligned}a = a \wedge (a \vee c) &= a \wedge (b \vee c) = b \vee (a \wedge c)\\ &= b \vee (b \wedge c) = b\end{aligned}$$

i.e., no interval can contain an element which has two *different* comparable complements relative to the interval.

Conversely, let the given condition holds. Suppose $\mathbf{L}$ is not modular. Then $\mathbf{L}$ contains a pentagonal sublattice

$$\{a \wedge c, b \vee (a \wedge c), a \wedge (b \vee c), b \vee c, c\}.$$

Put $a \wedge (b \vee c) = t$, $b \vee (a \wedge c) = r$, then $t > r$

Also $t \wedge c = r \wedge t = a \wedge c = x$ (say)

$t \vee c = r \vee t = b \vee c = y$ (say)

$x = t \vee c \leq t \vee c = y \Rightarrow x \leq y$

i.e., $\exists$ an interval $[x, y]$ in L which has an element c

$[x = t \wedge c \leq c \leq t \vee c = y]$,

with two *different* comparable complements t and r relative to $[x, y]$, (a contradiction.)

Hence **L** is modular. ■

Theorem 12: A complemented modular lattice is relatively complemented.

Proof: Let **L** be a complemented modular lattice.

Let $[a, b]$ be any interval in **L** and $x \in [a, b]$ be any element.

Since **L** is complemented, x has a complemented (say x')

Then $x \wedge x' = 0, x \vee x' = u, a \leq x \leq b.$

Taking $y = a \vee (b \wedge x')$

Then $x \wedge y = x \wedge [a \vee (b \wedge x')]$

$= a \vee (x \wedge (b \wedge x'))$ [as $x \geq a$, **L** is modular]

$= a \vee (b \wedge x \wedge x')$

$= a \vee (b \wedge 0) = a \vee 0 = a$

$x \vee y = x \vee [a \vee (b \wedge x')] = (x \vee a) \vee (b \wedge x')$

$= x \vee (b \wedge x')$

$= b \wedge (x \vee x')$ [as $x \leq b$, **L** is modular]

$= b \wedge u = b.$

Hence $y = a \vee (b \wedge x')$ is relative complement of x is $[a, b]$

Theorem 13: A chain is a distributive lattice

Proof: Let a, b, c be any three members of a chain, then any two of these are comparable.

Let $a \leq b, a \geq c, b \leq c$

then $a \leq b \leq c \leq a \Rightarrow a = b = c.$

Thus $a \wedge (b \vee c) = a = (a \wedge b) \vee (a \wedge c)$

if $a \leq b, a \geq c, c \leq b.$

then $c \leq a, a \leq b, c \leq b.$

thus $a \wedge (b \vee c) = a \wedge b = a.$

$(a \wedge b) \vee (a \wedge c) = a \vee c = a.$

we can change different cases ($a \leq b$, $a \leq c$, $a \leq c$, $a \geq b$, $a \geq c$) the condition of distributivity holds and thus a chain is always a distributive lattice. ■

Theorem 14: A lattice is distributive iff

$$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c), \forall\ a, b, c \in L.$$

Proof: Let **L** be distributive.

$$\begin{aligned}
\text{Now,} (a \vee b) \wedge (a \vee c) &= [(a \vee b) \wedge a] \vee [(a \vee b) \wedge c] \\
&= a \vee [(a \vee b) \wedge c] && \text{[absorption]} \\
&= a \vee [(a \wedge c) \vee (b \wedge c)] && \text{[distributivity]} \\
&= [a \vee (a \wedge c)] \vee (b \wedge c) && \text{[associativity]} \\
&= a \vee (b \wedge c)
\end{aligned}$$

conversely, let $a, b, c \in \mathbf{L}$ be any three elements, then

$$\begin{aligned}
(a \wedge b) \vee (a \wedge c) &= [(a \wedge b) \vee a] \wedge [(a \wedge b) \vee c] \\
&= a \wedge [(a \wedge b) \vee c] \\
&= a \wedge [(c \vee a)] \wedge [(c \vee b)] \\
&= [a \wedge (c \vee a)] \wedge (c \vee b) \\
&= a \wedge (c \vee b) = a \wedge (b \vee c)
\end{aligned}$$

i.e., **L** is distributive. ■

Note: Dual of a distributive lattice is distributive.

Theorem 15: A lattice **L** distributive iff

$$(a \vee b) \wedge (b \vee c) \wedge (c \vee a) = (a \wedge b) \vee (b \wedge c) \wedge (c \wedge a)\ \forall\ a, b, c \in L$$

Proof: Let **L** be a distributive lattice.

$$\begin{aligned}
(a \vee c) \wedge (b \vee c) \wedge (c \vee a) &= \{a \wedge [(b \vee c) \wedge (c \vee a)]\} \\
&\qquad \vee \{b \wedge [(b \vee c) \wedge (c \vee a)]\} \\
&= [\{a \wedge (c \vee a)\} \wedge (b \vee c)] \\
&\qquad \vee [\{b \wedge (b \vee c)\} \wedge (c \vee a)] \\
&= [a \wedge (b \vee c)] \vee [b \wedge (c \vee a)] \\
&= (a \wedge b) \vee (a \wedge c) \vee (b \wedge c) \vee (b \wedge a) \\
&= (a \wedge b) \vee (b \wedge c) \vee (c \wedge a).
\end{aligned}$$

Conversely, we first show that **L** is modular.

Let x, y, z be any three elements of **L**, with $x \geq y$.

Then $x \wedge (y \vee z) = [x \wedge (x \vee z)] \wedge (y \vee z)$ (absorption)

$= (x \vee y) \wedge (x \vee z) \wedge (y \vee z)$ $(x \geq y)$

$= (x \vee y) \wedge (y \vee z) \wedge (z \vee x)$

$= (x \wedge y) \vee (y \wedge z) \vee (z \wedge x)$

$= (y \vee (y \vee z)) \wedge (z \wedge x)$ $(x \geq y)$

$= y \vee (x \wedge z)$

i.e., **L** is modular.

Now for any $a, b, c \in \mathbf{L}$

$a \wedge (b \vee c) = [a \wedge (a \vee c)] \wedge (b \vee c)$

$= [a \wedge (a \vee b) \wedge (a \vee c)] \wedge (b \vee c)$

$= a \wedge [(a \vee b) \wedge (b \vee c) \wedge (c \vee a)]$

$= a \wedge [(a \wedge b) \vee (b \wedge c) \vee (c \wedge a)]$

$= a \wedge [(b \wedge c) \vee ((a \wedge b) \vee (c \vee a))]$

Now using modularity, as $a \geq a \wedge b$, $a \geq c \wedge a$ gives $a \geq (a \wedge b) \vee (c \wedge a)$

we get $a \geq c \wedge a$

$a \vee a \geq (a \wedge b) \vee (c \wedge a)$

$a \wedge (b \vee c) = [(a \wedge b) \vee (c \wedge a)] \vee [(b \wedge c) \wedge a)]$

$= (a \wedge b) \vee [(c \wedge a)] \vee [(c \wedge a) \wedge b]$

$= (a \wedge b) \vee (c \wedge a)$

Hence **L** is distributive. ■

Theorem 16: Homomorphic image of a distributive lattice is also distributive.

Proof: Let θ: $\mathbf{L} \to M$ be an onto homomorphism where L is a distributive lattice.

Let $x, y, z \in M$ be any element.

Since θ is onto, there exists

$a, b, c \in \mathbf{L}$ such that,

$\theta(a) = x, \theta(b) = y, \theta(c) = z$

Now $x \wedge (y \vee z) = \theta(a) \wedge [\theta(b) \vee \theta(c)]$

$= \theta(a) \wedge (\theta(b \vee c))$

$= \theta[a \wedge (b \vee c)]$

$= \theta((a \wedge b) \vee (a \wedge c))$

$$= \theta(a \wedge b) \vee \theta(a \wedge c)$$
$$= (\theta(a) \wedge \theta(b)) \vee (\theta(a) \vee \theta(c))$$
$$= (x \wedge y) \vee (x \wedge z)$$

Hence M is distributive lattice. ■

Theorem 17: The ideal lattice $I(\mathbf{L})$ of a distributive lattice **L** is distributive iff **L** is distributive.

Proof: Let **L** be distributive

Let $A, B, C \in I(\mathbf{L})$ be any three members, then A, B, C are ideals of **L**. We show

$$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$$

i.e, $$A \wedge (B \vee C) = (A \wedge B) \wedge (A \wedge C)$$

Let $x \in A \cap (B \wedge C)$ be any element.

Then $x \in A$ and $x \in B \vee C$

$\Rightarrow$ $\exists\, b \in B, c \in C$ s.t, $x \leq b \vee c$

$\Rightarrow$ $x \wedge (b \vee c) = x$

$\Rightarrow$ $(x \wedge b) \vee (x \wedge c) = x$ (as **L** is distributive)

Now $x \in A, b \in B \subseteq \mathbf{L} \Rightarrow x \wedge b \in A$

Again, $x \in A \subseteq \mathbf{L}$ and $b \in B \Rightarrow x \wedge b \in B$

$\Rightarrow$ $(x \wedge b) \in A \wedge B$.

Similarly, $(x \wedge c) \in A \wedge C$.

Since $x = (x \wedge b) \vee (x \wedge c)$, by definition of $\vee$ in $I(L)$

we find $x \in (A \wedge B) \wedge (A \wedge C)$

i.e., $A \wedge (B \vee C) \subseteq (A \wedge B) \wedge (A \wedge C)$

Again, let $x \in (A \wedge B) \wedge (A \wedge C)$ be any element.

Then $x \leq k_1 \vee k_2$ for some $k_1 \in A \wedge B$, $k_2 \in A \wedge C$.

Now $k_1 \vee k_2 \in A$, $x \leq k_1 \vee k_2$, thus $x \in A$.

Also $k_1 \in B$, $k_2 \in C$ and $x \leq k_1 \vee k_2 \Rightarrow x \in B \vee C$.

Thus $x \in A \wedge (B \vee C)$

or that $(A \wedge B) \vee (A \wedge C) \subseteq A \wedge (B \vee C)$

i.e., $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$

and hence $I(\mathbf{L})$ is distributive. ■

SOLVED EXAMPLES

Example 1

If $(X, \leq)$ is totally ordered, show that the lexicographics on $X \times X$ is also a total order.

Solution

Suppose that $(a, b), (c, d) \in X \times X$. As $\leq$ is a total order on X, for a and c, we have that either $a < c$ or $a = c$ or $c < a$.

If $a < c$, then we have $(a, b) < (c, d)$.

If $c < a$, then $(c, d) < (a , b)$

If $a = c$, then we compare b and d. We take $(a, b) < (c, d)$, $(a, b) = (c, d)$ or $(c, d) < (a , b)$ according as $b < d$, $b = d$, or $d < b$ holds.

In any case either $(a, b) \leq (c, d)$ or $(c, d) \leq (a, b)$

Hence $\leq$ is total order on $X \times X$. ■

Example 2

The lattice of subspaces of a vector space is modular, but is not distributive in general.

Solution

Let A, B, C be subspace of a vector space $\vee$ with $A \subseteq C$.

Then it suffices to show that

$$(A + B) \wedge C \subseteq A + (B \wedge C)$$

because the reverse inclusion holds modular inequality.

Now if $x \in (A + B) \wedge C$, then $x = (a + b) \in C$ with $a \in A$, $b \in B$,

Now $b = x - a \in C$ as $a \in A \subseteq C$ and $x \in c$.

$\therefore b \in B \wedge C$. $\therefore x = a + b \in A + (B \wedge C)$ which proves our claim.

To show that it is not distributive in general, take $\vee = \Re^2 = \{(x, y) : x, y \in \Re\}$, the two dimensional real vector space,

B = The X-axis, C = The Y-axis,

and $A = \{(x, x) : x \in X\}$ *i.e.,* the line $y = x$.

Then $(A \wedge (B + C)) = A \wedge V = A$.

while $(A \wedge B) + (A \wedge C) = (0) + (0) = (0)$.

and hence the subspace lattice of $\Re^2$ is not distributive. ■

Example 3

A lattice L is modular iff

$$a \wedge (b \vee (a \wedge c)) = (a \wedge b) \vee (a \wedge c) \; \forall \; a, b, c \in L.$$

Solution

Suppose that L is modular

$$a \leq c \Rightarrow a \vee (b \wedge c) = (a \vee b) \wedge c$$

Since $a \wedge c \leq a$, we have

$$(a \wedge c) \vee (b \wedge a) = [(a \wedge c) \vee b] \wedge a.$$

or $$(a \wedge b) \vee (a \wedge c) = a \wedge [b \vee (a \wedge c)]$$

Conversely given this condition for all *a, b, c* we write the above rule with a and c interchanged.

$$c[b \wedge (c \wedge a)] = (c \wedge b) \wedge (c \wedge a)$$

We now take $a \leq c$, so that $a \wedge c = a$.

$$c \wedge (b \vee a) = (c \wedge b) \vee a.$$

or $$a \vee (b \wedge c) = (a \vee b) \wedge c.$$

$\therefore$ The lattice is modular. ■

Example 4

Show that (D (105), g.c.d, l.c.m) is a uniquely complemented lattice.

Solution

Here $$D\,(105) = \{1, 3, 5, 7, 15, 21, 35, 105\}.$$

It may be readily checked that

$$1' = 105,\ 105' = 1.$$
$$3' = 35'\ 35' = 3.$$
$$5' = 21'\ 21' = 5.$$
$$7' = 15,\ 15 = 7.$$

In general in any bounded lattice, the bounds 0 and 1 are unique complements of each other because

$$0 \vee 1 = 1 \text{ and } 0 \wedge 1 = 0.$$

The uniqueness follows from the fact that if

$$0 \vee a = 1 \text{ and } 0 \wedge 1 = 0.$$

Then $$a = 0 \vee a = 1 \Rightarrow a = 1.$$

We recall that De Morgan's laws holds for complements of unions and intersections of subsets in the context of the power set lattice. ■

Example 5

Let N be the lattice of all natural numbers under divisibility. Show that $A = \{1, p, p^2, ...\}$ *where p is a prime, forms an ideal of N.*

Solution

A is clearly a non empty subset of N

Let $x, y \in A$ be any members. Then $x = p^n$, $y = p^m$ for some n. m $x \vee y = l.c.m\ (x, y) = l.c.m.\ (p^n, p^m) = p^k \in A$, when $k = \max(n, m)$.

Again, let $a \in A$ be any element and suppose $x \leq a$

Then $a \in A \Rightarrow a = p^n$ for some n

Also $x \leq a$ means $x|a \Rightarrow x = p^m$ for some $m \leq n$

$\Rightarrow x \in A$.

Hence A is an ideal.

SUMMARY

1. **Lattice** is a system of elements with two basic operations *viz.* meet and joint.
2. A **partial ordered** is a set which is Reflexive, Antisymmetric and Transitive.
3. The set of all statements form a poset under the ordering of tautological implication.
4. The set of all subgroups of a given group or that of all subspacas of a given space from poset under the relation of set inclusion.
5. A one-one onto map $f : P \to Q$ is called an isomorphism if ${}_xR_y \Leftrightarrow f(x)\ R'f(x)\ x, y \in P$. or if $x \leq y \Leftrightarrow f(x) \leq f(y)$
6. A map. $f : P \to$ is called **isotone** or order preserving if $x \leq y \Rightarrow f(x) \leq f(x)$.
7. Two posets are said to be **dually isomorphic** if one is isomorphic to the dual of the other.
8. A poset $(L, \leq)$ is said to be a *latter* if for every $a, b \in L$, *sub* $[a, b]$ and *inf* $[a, b]$ exists in L.

9. Dual of a lattice is a lattice.

10. Product of two lattices is a lattice.

11. A finite lattice has least and greatest elements.

12. A lattice is called complete iff every non-empty subset of L has its *sup* and *inf* in L.

13. Dual of a completed lattice is complete.

14. Every nonempty subset of a chain is a sublattice

15. Union of two sublattice may not be a sublattice.

16. An ideal is a sublattice but converse may not be true.

17. Intersection of two ideals in an ideal.

18. Union of two ideals is an ideal iff one of them is contained in the other

19. A dual ideal it a convex sublattice.

20. Intersection of dual ideals is a dual ideal.

21. Union of two dual ideal in a dual ideal iff one of them is contained in the other.

22. A sublattice of a modular lattice is modular.

23. A lattice of length two is modular.

24. Homomorphic image of modular lattice is modular.

25. A complemented modular lattice is relatively complemented.

26. A chain is a distributive lattice.

27. Dual of a distributive lattice is distributive.

28. Homomorphic image of a distributive

EXERCISE

1. Show that in a poset $a < a$ for no a and

$$a < b,\ b < c \Rightarrow a < c.$$

2. Let $X = \{1, 2, 3\}$ then show that $(P(X), \leq)$ is a poset.

3. A mapping $f : P \to Q$ is an isomorphism if and only if f is isotone and has isotone inverse.

4. Prove that, a poset $(L, \leq)$ is a lattice iff every non empty finite subset of L has sub. and inf.

5. Show that in a lattice L, the modular inequality $a \wedge (b \vee c) \geq b \vee (a \wedge c)$ holds for all $a, b, c \in L$, $a \geq b$.
6. Show that union of two sublattices may not be a sublattice.
7. Prove that, any interval $[a, b]$ in a lattice is a convex sublattice.
8. Show that
   (*i*) Finite chains are self dual.
   (*ii*) Chain N of natural under $\leq$ is not self dual.
   (*iii*) Chain Z of integers under $\leq$ is self dual.
9. Show that the union of two ideals may not be an ideal.
10. Show that an ideal of lattice L which is also a dual ideal is the lattice itself.
11. Show that, every convex sublattice of a lattice L is the intersection of an ideal and a dual ideal.
12. Prove that, in a finite lattice, every ideal is a principal.
13. Prove that, a sublattice of a modular lattice is a modular lattice.
14. Prove that, a lattice L is modular if and only if $I(L)$, the ideal lattice of L is modular.
15. If a and b are any elements of a modular lattice L then prove that $[a \wedge b, a] - [b, a \vee b]$.

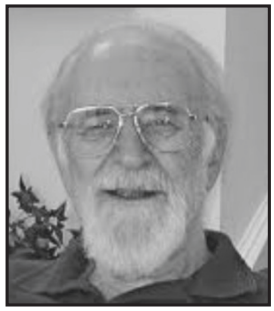
Suggested Readings

1. Thomas Donnellan, *Lattice Theory*, Khosla Publication, New Delhi.
2. H.A. Priestley, B.A. Davey, *Introduction to Lattices and order*, Combridge University Press, 2002.
3. D.A. Schmidt, *Denotational Sematics,* Allyn and Bacon, 1986.
4. A.S. Tanenbaum, Structured Computer Orgnization, 4th edition, PHI, 2010.
5. Vijay K. Khana, *Lattices and Boolean Algebras,* Vikas Publishing House, New Delhi 1985
6. S.K. Yadav, *Discrete Mathematical Structures*, Satya Prakashan, New Delhi; 2001.

❑❑❑

7

Boolean Algebras and Applications



Maurice Karnaugh

Maurice Karnaugh (1924-), is a physicist born in New York, USA. He graduated from the city college of New York in 1948. He received his Ph.D. in physics from Yale in 1952.

Karnaugh was a member of the research staff at Bell Telephone labs from 1952 to 1966. He was also a research and development manager at the Federal System Division of AT&T during the period 1966-1970. He joined IBM in 1970. His research interests include knowledge-based computer systems. His work on logic design and computer archetective is most popular. He developed the graphical method of the Karnaugh map at Bell laboratories in 1953 to simply the DNFs of Boolean expressions.

7.1 Introduction

Boolean Algebra was discovered by an English mathematician **George Boole** (1815–1864). His classic, *"An Investigation of the closely related area of mathematics"*, symbolic logic and the mathematical system called boolean algebra. Until the late 1930s boolean algebra did not seem to have many useful applications. In 1938 **Claude E. Shannon**, while working at the MIT used boolean algebra to analyze electrical circuits, thus opening the door for the world of applications of boolean algebra. Since then, boolean algebra has played a central role in the design, analysis and simplification of electronic devices including digital computers. The principle behind boolean algebra is that the answer to its equations are

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_7

either **True** or **False**. The digit 1 or letter '***T***' is used to represent a true solution and the digit 0 or letter '***F***' for false solution.

7.2 Boolean Algebra (Analytic Approach)

"*A boolean algebra is a lattice with 0 and 1 which is distributive and complemented.*"

We denote a Boolean algebra by $\mathbf{B} = (\mathbf{B}, \wedge, \vee, ', 0, 1)$ because it is bounded by definition of a complemented lattice. Thus **B** is a distributive bounded lattice in which every element a has a unique complement denoted by a'.

A boolean algebra is a set **B** together with two binary operations $\wedge$ and $\vee$, two binary operation 0 and 1 and unary operation of complementation satisfying the following properties:

1. $(\mathbf{B}, \wedge, \vee)$ is a lattice *i.e.*,

(*i*)	$a \wedge a = a$	(*i*)′	$a \vee a = a$
(*ii*)	$a \wedge b = b \wedge a$	(*ii*)′	$a \vee b = b \vee a$
(*iii*)	$a \wedge (b \wedge c) = (a \wedge b) \wedge c$	(*iii*)′	$a \vee (b \vee c) = (a \vee b) \vee c$
(*iv*)	$a \wedge (a \wedge b) = a$	(*iv*)′	$a \vee (a \vee b) = a$

for every $a, b, c \in \mathbf{B}$

2. The lattice $(\mathbf{B}, \wedge, \vee)$ is distributive *i.e.*

(*i*) $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)\ \forall\ a, b, c \in \mathbf{B}$

(*ii*) $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)\ \forall\ a, b, c \in \mathbf{B}$

(*iii*) $\{(a \wedge b) = (a \wedge c), (a \vee b) = (a \vee c)\} \Rightarrow \quad b = c\ \forall\ a, b, c \in \mathbf{B}.$

(*iv*) $(a \wedge b) \vee (b \wedge c) \vee (c \vee a) = (a \vee b) \wedge (b \vee c) \wedge (c \vee a)\ \forall\ a, b, c \in \mathbf{B}.$

3. $(\mathbf{B}, \wedge, \vee, 0, 1)$ is a bounded lattice *i.e.*, $\exists$ two distinct elements 0 and 1, s.t.

(*i*) $a \wedge 0 = 0, a \vee 0 = a$

(*ii*) $a \vee 1 = 1, a \wedge 1 = a$

holds for all $a \in \mathbf{B}$.

4. (**B**, $\wedge$, $\vee$, 0, 1) is a uniquely complemented lattice *i.e.* every element a has a unique complement a' satisfying;

   (*i*) $(a') = a, \forall\ a \in \mathbf{B}$

   (*ii*) $a \wedge a' = 0, a \vee a' = 1\ \forall\ a \in \mathbf{B}$

   (*iii*) $0' = 1, 1' = 0$

   (*iv*) $(a \wedge b)' = a' \vee b', (a \vee b') = a' \wedge b'\ \forall\ a, b \in \mathbf{B}$.

5. (**B**, $\leq$) is a poset with partial order $\geq$ satisfying:

   (*i*) $a \wedge b = glb\ \{a, b\}, a \vee b = \text{lub}\ \{a, b\}$

   (*ii*) $a \leq b \Leftrightarrow a \wedge b = a \Leftrightarrow a \vee b = b$

   (*iii*) $0 \leq a \leq 1\ \forall\ a \in \mathbf{B}$

   (*iv*) $a \leq b \Leftrightarrow a \wedge b' = 0 \Leftrightarrow a' \vee b = 1 \Leftrightarrow b' \leq a'$

Note: We have proved all the expressions above expect the very last one which can be proved by cyclic proof.

Let $a \leq b$.

$a \wedge b' \leq b \wedge b'$ (by, isotonicity)

Then $0 \leq a \wedge b' \leq b \wedge b' = 0$ gives $a \wedge b' = 0$ (by antisymmetry)

Now $a \wedge b' = 0$ gives $(a \wedge b') = 0'$ or $a' \vee b = 1$. (Complementation)

If $a' \vee b = 1$, then

$$a' = a' \vee 0 = a' \vee (b \vee b')$$
$$= (a' \vee b) \wedge (a' \vee b')' = 1 \wedge (a' \vee b') = a' \vee b'$$
$$b' \leq a' \text{ (by 5 (}ii\text{))}$$

finally if, $b' \leq a'$ is given, then $b' \vee a' = a'$

$\therefore$ $(a \wedge b)' = a'$ (by De-Morgan's law)

Thus, $a \wedge b = a$ (by 4(i) above)

$\therefore$ $a \leq b$.

For example: Let D_{30} be the set of positive factors of 30 such that $D_{30} = \{1, 2, 3, 5, 6, 10, 15, 30\}$. We may define three operations $\oplus$, $\odot$, and $'$ on D_{30} as follows:

$$a \oplus b = lcm\ \{a, b\}$$
$$a \odot b = gcd\ \{a, b\}$$
$$a' = \frac{30}{a}$$

For instance,

$$2 \oplus 3 = lcm\ \{2, 3\} = 6 = 3 \oplus 2;$$
$$6 \oplus 10 = lcm\ \{6, 10\} = 30 = 10 \oplus 6.$$
$$2 \odot 3 = gcd\ \{2, 3\} = 1 = 3 \odot 2;$$
$$6 \odot 10 = gcd\ \{6, 10\} = 2 = 10 \odot 6.$$
$$6' = \frac{30}{6} = 5 \quad 5' = \frac{30}{5} = 6.$$
$$6 \oplus 6' = 6 \oplus 5 = lcm\ \{6, 5\} = 30;$$
$$5 \odot 5' = 5 \odot 6 = gcd\ \{5, 6\} = 1$$

These operations follow the following properties:

(*i*) **Cummutativity,**

$$a \oplus b = b \oplus a,$$

(*ii*) **Associativity,**

$$a \oplus (b \oplus c) = (a \oplus b) \oplus c, \quad a \odot (b \odot c)$$
$$a \odot b = b \odot a = (a \odot b) \odot c$$

(*iii*) **Distributivity,**

$$a \odot (b \oplus c) = (a \odot b) \oplus (a \odot c),$$
$$a \oplus (b \odot c) = (a \oplus b)\ \odot (a \oplus c)$$

(*iv*) **Identity**

$$a \oplus 1 = a, \quad a \odot 30 = a$$

(*vi*) **Complementation,**

$$a \oplus a' = 30, \quad a \odot a' = 1.$$

7.2.1 Sub-Boolean Algebra

A Sub-Boolean Algebra of a given Boolean Algebra B is a subset B' of B, containing 0 and 1 equipped with induced lattice operations, which is closed under the lattice operations and complementation.

To check whether B' is a Boolean subalgebra of B, we need not actually check all of whether (*i*) B contains 0 and 1, (*ii*) is closed under $\wedge$, $\vee$ and complementation. Because of De Morgan's laws [$(a \wedge b)$ + $(a' \vee b')$] and $(a \vee b) = (a' \wedge b')$ and because of $1 = a \vee a'$, $0 = a \wedge a'$, it

suffices to check either (*i*) B' is closed under $\vee$ and $'$ *i.e.,* $a, b' \in B' \Rightarrow a \vee b, a' \in B'$ or (*ii*) B' is closed under $\wedge$ and $'$ *i.e.,* $a, b \in B' \Rightarrow a \wedge b, a' \in B'$.

7.2.2 Boolean Homomorphism

Let $(B_1, \wedge, \vee, 0, 1)$ and $(B_2, +, z, u, -)$ be two Boolean algebras.

A map. $f : B_1 \to B_2$ is called a Boolean Homomorphism if $f(a \wedge b) = f(a) * f(b); f(a \vee b) = f(a) \oplus f(b) \; \forall \; a, b \in B, f(0) = z, f(1) = u$ and.

$$f(a') = \overline{f(a)} \; \forall \; a \in B_1.$$

A *bijective homomorphism* is called a *boolean isomorphism*. It can be easily seen that it suffices to check the *f* preserves join and complements or that it preserves meet and complements.

Theorem 1 (Unique Identities): The zero element and the unique element of a boolean algebra B are unique.

Proof: The zero element in **B** will be proved unique.

Let there are two zero element 0_1 and 0_2 in **B**.

i.e. $\mathbf{B} = \{0_1, 0_2\}$

Since 0_2 is a zero element,

i.e. $0_1 + 0_2 = 0_1.$

Since 0_1 is a zero element,

hence $0_2 + 0_1 = 0_2.$

But $0_2 + 0_1 = 0_1 + 0_2$ (by commutativity)

$\Rightarrow$ $0_1 = 0_1 + 0_2 = 0_2 + 0_1 = 0_2.$

Hence zero element is unique.

(Second part can be done likewise)

Theorem 2 (Unique Complement): The complement of every element in a boolean algebra is unique.

Proof: Let *x* be an arbitrary element in a boolean algebra,

by complement laws,

$$x + x' = 1 \quad \text{and} \quad x\,x' = 0.$$

If *x* has a second complement *y*,

$$x + y = 1 \quad \text{and} \quad xy = 0.$$

To show that $y = x'$,

$$\begin{aligned} y &= y1 && \text{(identity law)} \\ &= y\,(x + x') && \text{(complement law)} \\ &= yx + yx' && \text{(distributive law)} \\ &= xy + x'y && \text{(commutative law)} \\ &= 0 + x'y && \text{(complement law)} \\ &= xx' + x'y && \text{(complement law)} \\ &= x'x + x'y && \text{(commutative law)} \\ &= x'\,(x + y) && \text{(distributive law)} \\ &= x'1 && \text{(complement law)} \\ &= x' && \text{(identity law)} \end{aligned}$$

Thus the complement of every element in a boolean algebra is unique.

Theorem 3: Let x and y be arbitrary element in a boolen algebra $\langle B, +, ., ' , 0, 1\rangle$. Then:

Idempotent laws

- $x + x = x$
- $xx = x$

Boundedness laws

- $x + 1 = 1$
- $x0 = 0$

Involution laws

- $(x')' = x$
- $0' = 1$
- $1' = 0$

Absorption laws

- $x + xy = x$
- $x\,(x + y) = x$

De Morgan's laws

- $(x + y)' = x'y'$
- $(xy)' = x' + y'$

Proof:

- **To prove that $x + x = x$:**

$$\begin{aligned} x + x &= (x + x)1 && \text{(identity law)} \\ &= (x + x)\,(x + x') && \text{(complement law)} \\ &= xx + xx' + xx + xx' && \text{(distributive law)} \end{aligned}$$

$= x + xx'$ (idempotent law)

$= x + 0$ (identity law)

$= x$ (identity law)

- **To prove that $x + 1 = 1$:**

$x + 1 = x + (x + x')$ (complement law)

$= (x + x) + x'$ (associative law)

$= x + x'$ (idempotent law)

$= 1$ (complement law)

- **To prove that $(x')' = x$:**

Since x' is the complement of x, $x + x' = 1$ and $xx' = 0$. Using the commutative laws, these equtions can be rewritten as

$$x' + x = 1 \quad \text{and} \quad x'x = 0 \qquad \ldots(i)$$

Since x' is also an element of B, it has a complement $(x')'$. Therefore,

$$x' + (x')' = 1 \quad \text{and} \quad x'(x')' = 0 \qquad \ldots(ii)$$

Equation (i) imply x is a complement of x'; by Equation (ii) $(x')'$ is also a complement of x'. But, by Theorem 2 the complement of x' is unique; therefore, $(x')' = x$.

- **To prove that $x + xy = x$:**

$x + xy = x1 + xy$ (identity law)

$= x(1 + y)$ (distributive law)

$= x(y + 1)$ (commutative law)

$= x1$ (boundedness law)

$= x$

- **To prove that $(x + y') = x'y'$:**

By the complement laws, we must show that $(x + y) + x'y' = 1$ and $(x + y)\,(x'y') = 0$.

1. $(x + y)\,x'y' = x + (y + x'y')$ (associative law)

$= x + (y1 + x'y')$ (identity law)

$= x + [y(x + x') + x'y']$ (complement law)

$= x + [(yx + yx') + x'y']$ (distributive law)

$= x + [xy + (x'y + x'y')]$ (commutative law)

$= (x + xy) + (x'y + x'y')$ (associative law)

$= x(1 + y) + x'(y + y')$ (distributive law)
$= x1 + x'(y + y')$ (boundedness law)
$= x1 + x'1$ (complement law)
$= x + x'$ (identity law)
$= 1$ (complement law)

2. $(x + y)(x'y') = (x'y')(x + y)$ (commutative law)
$= (x'y')x + (x'y')y$ (distributive law)
$= x(x'y') + (x'y')y$ (commutative law)
$= (xx')y' + x'(y'y)$ (associative law)
$= (xx')y' + x'(yy')$ (commutative law)
$= 0y' + x'0$ (complement law)
$= y'0 + x'0$ (commutative law)
$= 0 + 0$ (boundedness law)
$= 0$ (identity law)

Thus $x'y'$ is the complement of $x + y$; that is, $(x + y)' = x'y'$.

Theorem 4: *A Boolean Algebra is self dual.*

Proof: Let **B** be a Boolean algebra.

Define a map θ: $\mathbf{B} \to \mathbf{B}$, s.t

$$\theta(x) = x'$$

Then θ is well defined as for each $x \in \mathbf{B}$,

x' exists and is unique.

Now, $\theta(x) = \theta(y) \Rightarrow x' = y'$

$\Rightarrow$ $(x')' = (y')' \Rightarrow x = y$.

Thus θ is **one-one**, for any $y \in \mathbf{B}$, y' is its required pre image under θ showing that θ is **onto**.

Also $\theta(x \wedge y) = (x \wedge y)' = x' \vee y' = \theta(x) \vee \theta(y)$

$\theta(x \vee y) = (x \vee y)' = x' \wedge y' = \theta(x) \wedge \theta(y)$

shows that θ is a dual Homomorphism.

Thus θ is a dual isomorphism and hence **B** is self dual.

Theorem 5: In a Boolean Algebra, the following results hold.

(*i*) $(a')' = a$ (*ii*) $(a \wedge b)' = a' \vee b'$

(*iii*) $(a \vee b)' = a' \wedge b'$ (*iv*) $a \leq b \Leftrightarrow a' \geq b'$

(*v*) $a \leq b \Leftrightarrow a \wedge b' = o \Leftrightarrow a' \vee b = u$

Proof: (*i*) Let $(a')' = a''$ then

$$a \wedge a' = o,\ a \vee a' = u$$

$$a' \wedge a'' = o,\ a' \vee a'' = u$$

$$a \wedge a' = a'' \wedge a',\ a \vee a' = a'' \vee a'$$

$$\Rightarrow \qquad a'' = a$$

(*ii*) We have $(a \wedge b)' \wedge (a' \vee b')$

$$= [(a \wedge b) \wedge a'] \vee [(a \wedge b) \wedge b']$$

$$= [(a \wedge a') \wedge b'] \vee [(a \wedge b \wedge b')]$$

$$= [o \wedge b] \vee [a \wedge o] = o \vee o = o.$$

$$(a \wedge b) \vee (a' \vee b') = (a' \vee b') \vee (a \wedge b)$$

$$= [(a' \vee b') \vee a] \wedge [(a' \vee b') \vee b]$$

$$= [(a' \vee a) \vee b'] \wedge [(a' \vee (b' \vee b)]$$

$$= (u \vee b') \wedge (a' \vee u) = u \wedge u$$

$$= u \quad \Rightarrow \quad (a \vee b)' = a' \wedge b'$$

(*iii*) is similar as (*ii*)

(*iv*) $a \leq b \quad \Rightarrow \quad a = a \wedge b$

$\Rightarrow$ $a' = (a \wedge b)' = a' \vee b'$

$\Rightarrow$ $b' \leq a'$

$b' \leq a' \quad \Rightarrow \quad b'' \geq a'' \quad \Rightarrow \quad b \geq o.$

(*v*) $a \leq b \quad \Rightarrow \quad a \wedge b' \leq b \wedge b'$

$\Rightarrow$ $0 \leq a \wedge b' \leq o$

$\Rightarrow$ $a \wedge b' = o$

let $a \wedge b' = o$

$$a = a \wedge u = a \wedge (b \vee b') = (a \wedge b) \vee (a \wedge b')$$

$$= (a \wedge b) \vee o = (a \wedge b)$$

$$\Rightarrow \qquad a \leq a \wedge b.$$

Theorem 6: Let **B** be a bounded distributive lattice and $a \in \mathbf{B}$ then **B** can be imbedded into $[o, a] \times [a, u]$

Proof: We define a map $\theta: \mathbf{B} \to [o, a] \times [a, u]$, s.t.,

$$\theta(x) = (x \wedge a,\ x \vee a)$$

clearly then $x \wedge a \in [0, a]$ and $x \vee a \in [a, u]$.

Let $x = y \Rightarrow \quad x \wedge a = y \wedge a$

$$x \vee a = y \vee a$$

$\Rightarrow \quad (x \wedge a, x \vee a) = (y \wedge a, y \vee a).$

$\Rightarrow \quad \theta(x) = \theta(y).$

Thus θ is well defined.

Again, if $\theta(x) = \theta(y)$

then $(x \wedge a, x \vee a) = (y \wedge a, y \vee a)$

$\Rightarrow \quad x \wedge a = y \wedge a$

$x \vee a = y \vee a$

$x = y$

Thus θ is **one-one**

Now $\theta(x \wedge y) = ((x \wedge y) \wedge a, (x \wedge y) \vee a)$

$= ((x \wedge a) \wedge (y \wedge a), (x \vee a) \wedge (y \vee a))$

and $\theta(x) \wedge \theta(y) = (x \wedge a, x \vee a) \wedge (y \wedge a, y \vee a)$

$= ((x \wedge a) \wedge (y \wedge a), (x \vee a) \wedge (y \vee a))$

shows that $\theta(x \wedge y) = \theta(x) \wedge \theta(y)$

similarly $\theta(x \vee y) = \theta(x) \vee \theta(y)$

Hence θ is a 1-1 homomorphism, *i.e.,* θ is an imbedding map.

Theorem 7: If **B** is a Boolean algebra and $a \in \mathbf{B}$, then $\mathbf{B} \cong [o, a] \times [a, u]$.

Proof: By previous theorem

$\theta : \mathbf{B} \to [o, a] \times [a, u]$ s.t.,

$\theta(x) = (x \wedge a, x \vee a)$

is a 1-1 homomorphism. (Notes–a Boolean algebra is distributive). We show θ is onto.

Let $(y, z) \in [o, a] \times [a, u]$ be any element

then $o \leq y \leq a, \quad a \leq z \leq u$ …(*i*)

Take $x = y \vee (z \wedge a')$, then

$\theta(x) = \theta(y \vee (z \wedge a'))$

$= ((y \vee (z \wedge a')) \wedge a, y \vee (z \wedge a') \vee a)$

$= ((y \wedge a) \vee ((z \wedge a') \wedge a), y \vee ((z \vee a) \wedge (a' \vee a)))$

$= (y \vee (z, \wedge o), y \vee (z \wedge u))$

$= (y, z)$ [From (*i*)]

Hence θ is an isomorphism.

Theorem 8: If A, B, C are lattices such that $B \cong C$, then $A \times B \cong A \times C$.

Proof: Let $f: B \to C$ be the given isomorphism,

we define $\theta: A \times B \to A \times C$, s.t.

$$\theta((a, b)) = (a, f(b))$$

Then since, $\theta((a, b)) = \theta((c, d))$

$\Leftrightarrow$ $(a, f(b)) = (c, f(d))$

$\Leftrightarrow$ $a = c, f(b) = f(d)$

$\Leftrightarrow$ $a = c, b = d$

(f being well defined **one-one** map)

$\Leftrightarrow$ $(a, b) = (c, d)$

we find that θ is a well defined one-one map.

Again, for any $(x, y) \in A \times C$, as $y \in c, f: B \to C$ is **onto**

There exists $b \in B$ s.t. $f(b) = y$.

then $\theta((x, b)) = (x, f(b)) = (x, y)$ and thus θ is **onto**

Finally,

$$\theta((a, b) \wedge (c, d)) = \theta(a \wedge c, b \wedge d) = (a \wedge c, f(b \wedge d))$$
$$= (a \wedge c, f(b) \wedge f(d)) = (a, f(b)) \wedge (c, f(d))$$
$$= \theta((a, b)) \wedge \theta(c, d))$$

Similarly,

$$\theta((a, b) \vee (c, d)) = \theta((a, b)) \vee \theta((c, d))$$

Hence θ is an **isomorphism.**

Theorem 9. If **B** is a Boolean algebra and $a \in \mathbf{B}$, then show that $B \cong [0, 1] \times [0, a']$

Proof. By theorem 7, we have

$$L \cong [0, a] \times [a, u]$$

we define a map. $f: [a, u] \to [a, a']$. Such that

$$f(x) = x \wedge a'$$

Now $x \in [a, u] \Rightarrow a \leq x \leq u$

$\Rightarrow$ $a \wedge a' \leq x \wedge a' \leq u \wedge a'$

$\Rightarrow$ $0 \leq x \wedge a' \leq a' \Rightarrow x \wedge a' \in [0, a']$

and as $x = y \Rightarrow x \wedge a' = y \wedge a' \Rightarrow f(x) = f(y)$

f is well defined.

Again $f(x) = f(y)$

$\Rightarrow$ $x \wedge a' = y \wedge a'$

$\Rightarrow$ $(x \wedge a') \vee a = (y \wedge a') \vee a$

$\Rightarrow$ $(x \vee a) \vee (a' \vee a) = (y \vee a) \wedge (a' \vee a)$

$\Rightarrow$ $x \wedge u = y \wedge u \Rightarrow x = y$

Thus f is **one-one**

Now $f(x \wedge y) = (x \wedge y) \wedge a' = (x \wedge a') \wedge (y \wedge a')$

$= f(x) \wedge f(y)$

$f(x \vee y) = (x \vee y) \wedge a' = (x \wedge a') \vee (y \wedge a')$

$= f(x) \vee f(y)$

Hence f is a homomorphism.

Finally, let $y \in [o, a']$ be any element.

Then $o \leq y \leq a'$

$\Rightarrow$ $a \vee o \leq a \vee y \leq a \vee a'$ or $a \leq a \vee y \leq u$

$\Rightarrow$ $a \vee y \in [a, u]$

and as $f(a \vee y) = (a \vee y) \wedge a'$

$= (a \wedge a') \vee (y \wedge a') = o \vee y = y$

we find f is onto and hence an isomorphism, we get

$$B \cong [o, a] \times [a, u] \cong [o, a] \times [o, a']$$

Lemma: Let **B** be a finite Boolean algebra (with more than one element). Let $x \in$ **B** be a non zero element and suppose $a_1, a_2, ..., a_r$ are the atoms in **B** s.t., $a_i \leq x$ then $x = a_1 \vee a_2 \vee ... \vee a_r$ and this representation is unique.

Proof: Since $a_1 \leq x, a_2 \leq x, ..., a_r \leq x$

we get $a_1 \vee a_2 \vee ..... \vee a_r \leq x \vee x \vee ... \vee x = x.$

Put $a_1 \vee a_2 \vee ... \vee a_r = y.$

We claim $x \wedge y' = o.$

Suppose $x \wedge y' \neq o$, *i.e.*, $x \wedge y' > o$

Thus $\exists$ an atom a s.t., $o < a \leq x \wedge y'$

$\Rightarrow$ $a \leq x, y'$

$a \leq x$ means $a = a_i$ for some i

thus $a = a_i \leq a_1 \vee a_2 \vee ..... \vee a_r = y$

So $a \leq y', a \leq y \Rightarrow a \leq y \wedge y' = o$

which is not possible as $a > o$.

Thus $x \wedge y' = o$

$\Rightarrow \qquad x \leq y$

Hence $\qquad x = y = a_1 \vee a_2 \vee ... \vee a_r.$

To show uniqueness of this representation

Suppose $x = b_1 \vee b_2 \vee ... \vee b_k$ where b_j are also atoms.

Then $\qquad b_j \leq x \quad \forall\, j = 1, 2, \ldots, k$

$\Rightarrow \qquad b_j \wedge x = b_j$

$\Rightarrow \qquad b_j \wedge (a_1 \vee a_2 \vee ... \vee a_r) = b_j$

$\Rightarrow \qquad (b_j \wedge a_1) \vee (b_j \vee a_2) \vee ... \vee (b_j \wedge a_r) = b_j > o$

$\Rightarrow$ at least one $(b_j \wedge a_i) \neq o$.

But b_j and a_i are both atoms thus $b_j = a_i$

i.e., each b_j is some $a_i \Rightarrow$ uniqueness. ■

Theorem 10. Let **B** be a finite Boolean algebra and let S = set of all atoms of **B**, then $\mathbf{B} \cong P(S)$

Proof: Let us define a map $\theta : \mathbf{B} \to P(S)$, s.t.,

$$\theta(o) = \varphi$$

$$\theta(x) = \{a_1, a_2, ..., a_r\}$$

where a_i are atoms s.t., $\quad a_i \leq x \;\; (x \neq o)$.

Then θ is well defined in view of the above lemma and also then

$$x = a_1 \vee a_2 \vee ... \vee a_r$$

Now, $\qquad \theta(x) = \theta(y)$

$\Rightarrow \qquad \{a_1, a_2, ..., a_r\} = \{b_1, b_2, ..., b_s\} \quad a_i \leq x$

$$b_j \leq y$$

$\Rightarrow \qquad r = s$ and $a_i = b_j$

$\Rightarrow \qquad r = s$ and $a_i = b_i$ (without loss of generality)

$\Rightarrow \qquad x = a_1 \vee a_2 \vee ... \vee a_r$

$$= b_1 \vee b_2 \vee ... \vee b_s = y$$

$\Rightarrow \qquad \theta$ is one-one.

For any member $\{a_1, a_2, ..., a_r\}$ of $P(S)$, let $x = a_1 \vee a_2 \vee ... \vee a_r$ then x is the required pre image showing that θ is onto.

Consider now, $\theta\,(x \wedge y)$, for $x, y \in \mathbf{B}$

Let $\theta(x \wedge y) = \{c_1, c_2, ..., c_t\}$ where $c_i \leq x \wedge y$

then $c_i \leq x, c_i \leq y$

and $x \wedge y = c_1 \vee c_2 \vee ... \vee c_t$

If $\theta(x) \vee \theta(y) = \{a_1, a_2, ..., a_r\} \cap \{b_1, b_2, ..., b_s\}$

then $a_i \leq x, b_j \leq y, x = a_1 \vee a_2 \vee ... \vee a_r, y = b_2 \vee b_2 \vee ... \vee b_s$

Suppose $\{a_1, a_2, ..., a_r\} \cap \{b_1, b_2, ..., b_s\} = \{d_1, d_2, ..., d_k\}$

then $d_i \leq x$, $d_i \leq y$ and so $d_i \leq y$

i.e., each d_i is some c_j.

Again as each $c_i \leq x$, it is some a_i

and $c_i \leq y$, it is some b_j.

In other words, each c_i is some d_i showing that $\theta(x \wedge y) = \theta(x) \wedge \theta(y)$

Similarly, we can say $\theta(x \vee y) = \theta(x) \vee \theta(y)$ and hence θ is an isomorphism.

Cor. Since $o(P(S)) = 2^n$ where $o(S) = n$

we find $o(B) = 2^n$ where n = member of atoms that **B** has.

7.3 Boolean Functions

A Boolean variable assumes the value 0 or 1. A boolean function is a function $f: B^n \to B$, where $B^n = B \times B \times B \times ... \times B$, the cartisian product of B with itself to n factors.

Let $(x_1, x_2, x_3, ..., x_n) \in B^n$.

Then the values of the boolean function f at $(x_1, x_2, x_3, ..., x_n)$ is denoted by $f(x_1, x_2, x_3, ..., x_n\}$, although technically it should be $f((x_1, x_2, x_3, ..., x_n))$.

Since codom $(f) = B$, the value of the boolean function for any input value $(x_1, x_2, x_3, ..., x_n)$ is other 1 or 0.

Boolean functions can be defined by logic tables define two boolean functions from B^2 to B.

The value $f(x, y)$ is given for every combination value of x any y.

Boolean functions can also be defined by boolean expression made up of boolean variables and boolean operators. For instance, if x, y and z are boolean variables, then xy, $(xy)'$, $x + yz$, and $x + (xy)'$ are boolean expressions.

x	y	$f(x, y)$
0	0	0
0	1	1
1	0	1
1	1	1

x	y	$f(x, y)$
0	0	0
0	1	0
1	0	0
1	1	1

Fig. 7.1. Logic tables.

Let $x_1, x_2, x_3, ..., x_n$ be n boolean variables. Then.

Basis Clause: $0, 1, x_1, x_2, ..., x_n$ are boolean expressions.

Recursive Clause: If E_1 and E_2 are boolean expressions, then so are (E_1), E_1', E_1E_2, and $E_1 + E_2$.

7.3.1 Equality of Boolean Expressions

Two boolean expressions $E_1(x_1, x_2, x_3, ..., x_n)$ and $E_2(x_1, x_2, x_3, ..., x_n)$ are **equal**, denoted by $E_1 (x_1, x_2, x_3..., x_n) = E_2(x_1, x_2, x_3, ..., x_n)$ if they yield the same value for all $(x_1, x_2, x_3, ..., x_n)$ in B^n. When the boolean expressions E_1 and E_2 are simple, logic tables can determine if they are equal.

7.3.2 Minterms and Maxterms

A minterm in n variables $x_1, x_2, x_3, ..., x_n$ is a boolean expression on $y_1y_2y_3...y_n$ where each y_i is either x_i or x_i'. A literal is a boolean variable or its complement. Thus each y_i is a literal, and minterm is the product of n literals. A minterm contains a literaly corresponding to each variable x_i, y_i is either x_i or x_i'. In two variables, four minterms are possible; xy, xy', $x'y$ and $x'y'$.

The Boolean expressions xyz, xyz' and $xy'z$ are a few minterms in three variables.

Definition: A Boolean expression in n variables

$x_1, ..., x_n$ of the form.

$x_1\alpha_1 \wedge ... \wedge x^{\alpha}n^t$

where each $\alpha_i = 0$ or 1 with $x_i^1 = x_i$ and $x_{10} = x'_i$.

for $i = 1, 2, ...,$ is called a minterm, or complete product in these n variables. Observe that each minterm is completely determined by a sequence of 0s and 1s of length n, and any such sequence determines a number between 0 and $2^n - 1$ in binary representation.

A particular minterm will be denoted by $\min_j$ or m_j if the associated sequence of its exponents gives the number j in binary representation (Here $0 \le j \le 2^n - 1$). We thus have 2^n minterms in n variables denoted by $m_0, m_1, ..., m_{2n-1}$. For example in three variables.

$m_5 = m'_1 \wedge x_2^0 \wedge x'_3 = = x_1 \wedge x'_2 \wedge x_3$ because 5 has the binary representation 101.

These minterms satisfy following fundamental properties.

(*i*) $m_i \wedge m_i = 0$ if $i \neq j$.

(*ii*) $\bigvee_{i=0}^{2^n-1} m_i = m_0 \vee m_1 \vee ... \vee m_{2n-1}$.

Accordingly we have for the maxterms.

(*i*) $\max_i \vee \max_j = 1$ if $i \neq j$.

(*ii*) $\bigvee_{i=0}^{2^n-1} \max_j = 0$.

7.3.3 Functional Completeness

A set of boolean operators is called functionally complete if every boolean function can be defined using then.

The triad of boolean operators {+, ·, ′} forms a functionally complete set, but can fewer operators do so?

By De-Morgan's law,

$$(x + y)' = x'y', \text{ so } x + y = [(x + y)']' = (x'y')'$$

consequently the boolean operator '+' can be defined using the operators '.' and ''. Thus {. , ′} is a functionally complete set of two boolean operators.

7.3.4 NAND and NOR

The binary operators NAND (not AND) and NOR (not OR), represented by ↑ and ↓ respectively, are defined as follows.

$$x \uparrow y = \begin{cases} 0, & \text{if } x = 1 = y \\ 1, & \text{otherwise} \end{cases}$$

$$x \downarrow y = \begin{cases} 1, & \text{if } x = 0 = y \\ 0, & \text{otherwise} \end{cases}.$$

SOLVED EXAMPLES

Example 1

$B_2^n = B_2 \times B_2 \times ... \times B_2$ (in copies) then prove that $B_2^n = \{(a_1, a_2, ..., a_n\}$: $a_1 \in [0, 1]\}$

Solution

We define the lattice operations component wise.

$$(a_1, ..., a_n) \vee (b_1, ..., b_n) = (a_1 \vee b_1, ..., a_n \vee b_n)$$

$$(a_i, ..., a_n) \wedge (b_1, ..., b_n) = (a_1 \wedge b_1, ..., a_n \wedge b_n)$$

Here $0 \vee 0 = 0, 0 \vee 1 = 1 \vee 0 = 1 \vee 1 = 1$

$0 \wedge 0 = 0 \wedge 1 = 1 \wedge 0 = 0, 1 \wedge 1 = 1.$

It can be seen that the boundary (0, 0, ..., 0) and (1, 1, ..., 1), $(a_1, ..., a_n) \leq (b_1, ..., b_n)$ if $(a_i = 1 \Rightarrow b_i = 1)$ and $(a_1, ..., a_n)'$ is obtained by replacing 0's by 1's by 0's in $(a_1, ..., a_n)$.

Note: This algebra is called the **switching algebra.** ■

Example 2

The divisor lattice $D(n)$ of a positive integer $n > 1$ is a Boolean Algebra iff n is square free than it is product of distinct primes.

Solution

$D(n)$ is a bounded distributed lattice. (**Complementedness**)

If $n = p_1, p_2, ..., p_n$ is a product of distinct primes $p_1, p_2, ..., p_m$, then any divisor d of n is product of some of the primes from the list (above)

If $d = p_{i1}, ..., p_{ik}$ then take $d' = p_{j1}, ..., p_{js}$ be the product of all the prime divisors of n not dividing d *i.e.,* $(p_{j1}, ..., p_{rs}) = (p_1, ..., p_m) - (p_{i1}, ..., p_{im})$.

Then clearly g.c.d. $(d, d') = 1$ and 1.c.m. $(d, d') = \dfrac{dd'}{g.c.d.(d,d')}$

$= dd' = n$.

$\therefore$ d' is the complement of d. Thus $D(n)$ is complemented and so it is Boolean Algebra.

Conversely, if some $p^\alpha \mid n$ with $\alpha > 1$, then a particular the prime divisor p cannot have a complement in $D(n)$ because if $(p, d) = 1$ for some $d \in D(n)$, then p does not divide; $d \quad \therefore \ (p^\alpha, d) = 1$.

Thus $p^{\alpha} \wedge d = [p^{\alpha}, d] \times (p^{\alpha} \times d) = [p^{\alpha}, d] \leq n.$

Hence $p \,.\, d < p^{\alpha} \,.\, d \leq n.$ *i.e.,* $p \,.\, d < n.$

Then $$[p, d] = \frac{p.d}{(p.\, d)} = pd < n$$

and so d cannot be a complement of p. So p does not have a complement in $D(n)$ and so $D(n)$ is not a Boolean Algebra. ■

Example 3

Show that if B is a Boolean Algebra and $\phi \neq B' \subseteq B$, *then* B' *is a Boolean subalgebra of B if*

(i) $a, b \in B' \Rightarrow a \wedge b \in B'$

(ii) $a \in B' \Rightarrow a'\ b'$.

Solution

Suppose that B' satisfying the condition (i) and (ii). By hypothesis, there exists $a \in B'$, then by (ii), $a' \in B'$. By (i) we get

$$0 = a \wedge a' \in B'$$

$\therefore$ by (ii) $1 = 0' \in B'$, finally if $a, b \in B'$, then

$a \vee b\ (a' \wedge b') \in B'$ by De Morgan's law. ■

Example 4

Show that if $f : B_1 \to B_2$ *satisfies*

$$f(a \wedge b) = f(a) * f(b), f(0) = z$$

and $$f(a') = \overline{f(a)}\ \forall\ a, b \in B_1$$

then f is a Boolean homomorphism.

Solution

By hypothesises $f(1) = f(0') = \overline{f(a)} = \bar{z} = u$

$$f(a \vee b) = f(a' \wedge b') = f(a' \wedge b')$$

$$= \overline{f(a') * f(b')} = \overline{f(a)} * \overline{f(b)}$$

$$= f(a) \oplus f(b)\ \forall\ a, b \in B_1$$

$\therefore$ f is a Boolean homomorphism. ■

Example 5

With the Boolean algebra D_{30}, verify each of the following

1. $6 \oplus 6 = 6$
2. $5 \oplus 30 = 30$
3. $(5')' = 5$
4. $3 \oplus (3 \odot 5) = 3$
5. $(3 \oplus 5)' = 3' \odot 5'$
6. $(5 \odot 6)' = 5' \oplus 6'$

Solution

1. $6 \oplus 6 = \text{lcm } \{6, 6\} = 6$
2. $5 \oplus 30 = \text{lcm } \{5, 30\} = 30$
3. $5' = \dfrac{30}{5} = 6$

   So $(5')' = 6' = \dfrac{30}{6} = 5$
4. $3 \odot 5 = \text{gcd } \{3, 5\} = 1$

   So $3 \oplus (3 \odot 5) = 3 \oplus 1$
   $= \text{lcm } \{3, 1\} = 1$
5. $3 \oplus 5 = \text{lcm } \{3, 5\} = 15$

   So $(3 \oplus 5)' = 15' = \dfrac{30}{15} = 2$

   $3' \odot 5' = 10 \odot 6$
   $= \text{gcd } \{10, 6\} = 2$
   $= (3 \oplus 5)'$
6. $5 \odot 6 = \text{gcd } \{5, 6\} = 1$

   So $(5 \odot 6)' = 1' = \dfrac{30}{1} = 30$

   $= 5 \oplus 6$
   $= \text{lcm } \{5, 6\}$
   $= 5' \oplus 6'$ ■

Example 6

Evaluate the boolean expression $x + (yz)'$ for the triplets (0, 1, 0) and (0, 1, 1).

Solution

When $x = 0, y = 1, z = 0.$

$\Rightarrow$ $x + (yz)' = 0 + (1.0)' = 0 + 0' = 0 + 1 = 1.$

when $x = 0, y = 1, z = 1.$

$\Rightarrow$ $x + (yz)' = 0 + (1.1)' = 0 + 1' = 0 + 0 = 0.$ ■

Example 7

Using a logic table, verify that $(x + y)' = x'y'$.

Solution

As in propositional logic, constructing Fig. 7.2 shows that the columns headed by $(x + y)'$ and $x'y'$ are identical; therefore, $(x + y)' = x'y'$.

x	y	$x+y$	$(x+y)'$	x'	y'	$x'y'$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

Identical columns

Fig. 7.2. ■

Example 8

Construct a logic table for the boolean function

$$f(x, y, z) = (x + y + z)\,(xyz)'$$

Solution

Just as we build up truth tables for complex propositions from simple ones, we build up the boolean expression $(x + y + z)\,(xyz)'$ from boolean subexpressions in steps: x, y, z, $x + y$, $x + y + z$, xy, xyz, $(xyz)'$, and finally $(x + y + z)\,(xyz)'$. The table contains a column for each subexpressions; fill in the various columns using the definitions of sum, product, and complement. Fig. 7.3 is the resulting table.

x	y	z	$x+y$	$(x+y)+z$	xy	xyz	$(xyz)'$	$f(x, y, z) = (x+y+z)\,(xyz)'$
0	0	0	0	0	0	0	1	0
0	0	1	0	1	0	0	1	1
0	1	0	1	1	0	0	1	1
0	1	1	1	1	0	0	1	1
1	0	0	1	1	0	0	1	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	1	1
1	1	1	1	1	1	1	0	0

Fig. 7.3. ■

Example 9

Construct the DNF of the boolean function f (x, y, z) = x (y + z).

Solution

First, set up the logic table for the boolean function *f*. (See Fig. 7.4.) As before, find a minterm corresponding to each value 1 of the function and add the minterms to produce the DNF:

$$f(x, y, z) = xyz + xy'z + xyz'$$

x	y	z	$x + z$	$f(x, y, z) = x(y + z)$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Fig. 7.4. ■

Example 10

Verify that {↑} is functionally complete.

Solution

Since {., ′} is functionally complete, it suffices to show that both operators . and ′ can be expressed in terms of ↑. First, notice that

$$(xy)' = x \uparrow y \qquad ...(i)$$

So $(xx)' = x \uparrow x$

But $xx = x$, by the idempotent law; so

$$x' = x \uparrow x \qquad ...(ii)$$

By the involution law, $xy = [(xy)']'$

$= (xy)' \uparrow (xy)'$ by Equation (ii)

$= (x \uparrow y) \uparrow (x \uparrow y)$ $...(iii)$

which completes the proof.

7.4 Combinatorial Circuits (Synthesis of Circuits)

Logic gates can be used to construct combinatorial circuits for a number of applications in electrical and Electronics circuits in digital form. A combinatorial circuit is memoryless, but can produce a unique output for every combination of output signals. The output does not depend on previous inputs or the state of the system.

For example,

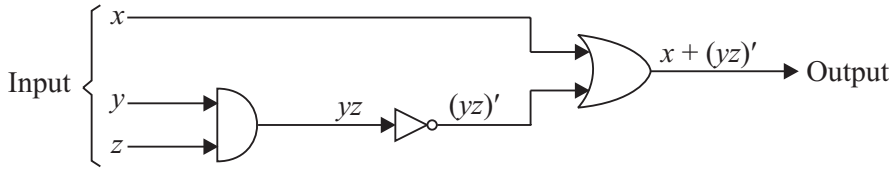


Fig. 7.5.

The output from the AND gate is yz. It enters an inverter, which outputs (yz'). Then x and $(yz)'$ are input to an OR gate, for the final output of $x + (yz)'$.

7.4.1 Half-Adder and Full-Adder

Suppose we would like to design a circuit to add any two bits x and y. From Fig. 7.6 we can see that $c = xy$ and $s = xy' + x'y$ (the DNF of s). But $xy' + x'y = (x + y)\,(xy)'$

so $s = (x + y)\,(xy)'$.

As a result, the circuit that yields the sum bit s and the carry bit c lies in the Figure 7.6. Being a **Half-adder**, it adds only two bits without considering the carry from a possible previous addition. It contains two AND gates, an OR gate and a NOT gate.

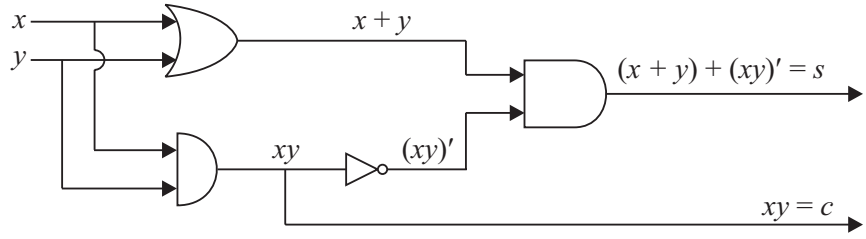


Fig. 7.6.

Full-adders find the sum of any two binary numbers. A **full-adder** accepts three bits; the two bits x_i and y_i in the addends and the carry bit c_i, where $i \geq 1$. After accepting these three bits, the full adder outputs the sum bit s_i and new carry bit c_{i+1} (As shown in Fig. 7.7). The values of s_i and c_{i+1} for various values of x_i, y_i and c_i.

According to the table,

$$s_i = x_i y_i c_i + x_i y_i' c_i' + x_i' y_i c_i' + x_i' y_i' c_i$$

and

$$c_{i+1} = x_i y_i c_i + x_i y_i c_i' + x_i y_i' c_i + x_i' y_i c_i$$

Consequently, a full adder can be made with AND, OR and NOT gates. Two half-adders will work well together as a full-adder. To find the sum bit s_i and the carry bit c_{i+1}, inputs x_i and y_i into a half-adder H_1, it produces a sum bit s and a carry bit c. The bits s and c_i are then input to another half-adder H_2. Their sum is the desired sum bit s_i. The carry bits from H_1 and H_2 are sent to an OR gate, which outputs the new carry bit c_{i+1}. This full-adder appears in Fig. 7.7 below

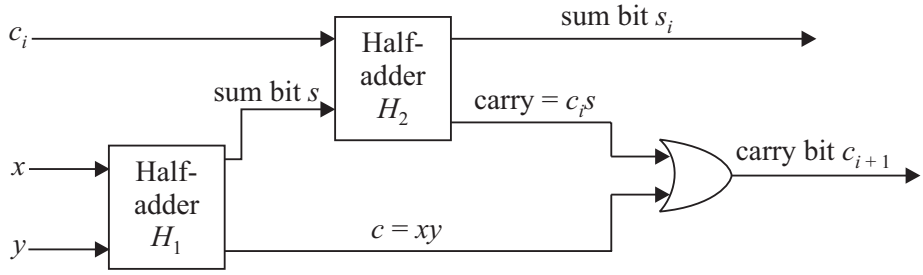


Fig. 7.7.

Input			Output	
x_i	y_i	c_i	s_i	c_{i+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Fig. 7.8.

For example: Using a half-adder and full-adder, design a circuit that computes that sum of two 3-bit numbers $x = x_2x_1x_0$ and $y = y_2y_1y_0$.

A half-adder accepts two bits outputing the sum and carry bits. On the other hand, a full-adder accepts three bits to produce the sum and carry bits. The circuit in Fig. 7.9 produces the sum $s = s_3s_2s_1s_0$.

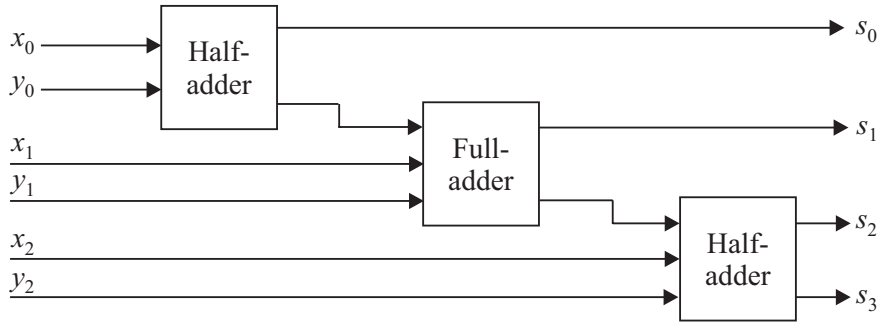


Fig. 7.9.

7.4.2 Equivalent Combinatorial Circuits

Two combinatorial circuits are said to be equivalent if the corresponding boolean expressions are equal.

We can illustrate this property through the following example.

In the above circuit (Fig. 7.10) we want to show whether circuits C_1 and C_2 are equivalent. To prove this, first, find the boolean expression representing the two circuits, $E_1 = x'y'$ and $E_2 = (x + y)'$ respectively.

Combinatorial circuit C_1 Combinatorial circuit C_2

Fig. 7.10.

Since $E_1 = E_2$ by De-Morgan's Law, the two combinatorial circuits are equivalent.

Circuit C_1 contains three gates; C_2, only two. Accordingly, C_2 is **simpler** than C_1.

Through two methods, algebraic and graphical, we can find a circuit C_2 equivalent to but simpler than a given circuit C_1. Since a gate in a

combinatorial circuit corresponds to a boolean operator, the key to both procedures lies in locating a boolean expression with fewer boolean operators and literals.

SOLVED EXAMPLES

Example 1

Represent the following circuit as a gating network and calculate the corresponding Boolean expression.

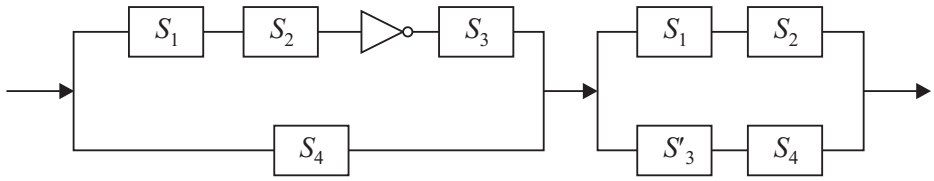


Fig. 7.11.

Solution

The equivalent gating network is

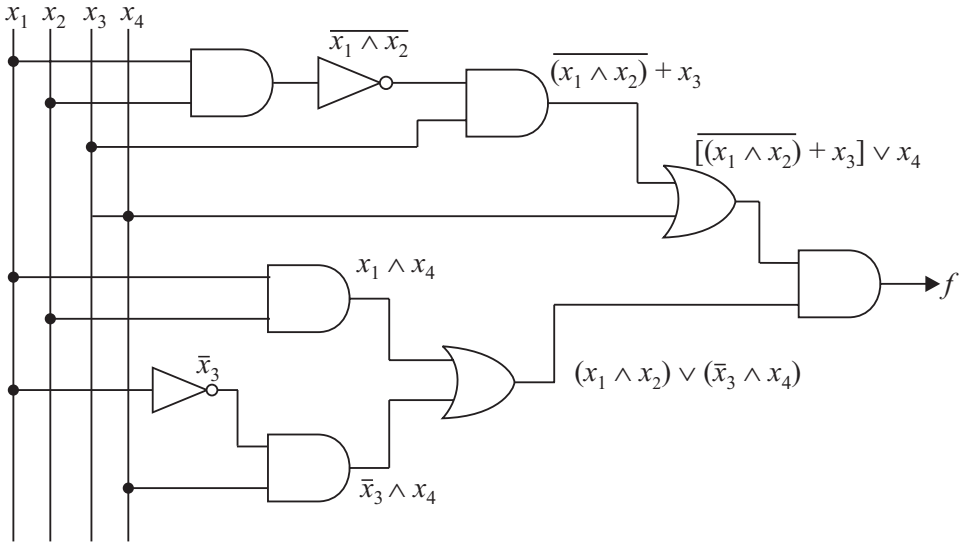


Fig. 7.12.

The Boolean expression for the output clearly is $f(x_1, x_2, x_3, x_4) = [\{x_1, \wedge x_2) \wedge x_3\} \vee x_4] \wedge \{x_1 \wedge x_4\}]$ ■

Example 2

Find a simpler combinatorial circuit equivalent to the one in Figure 7.13

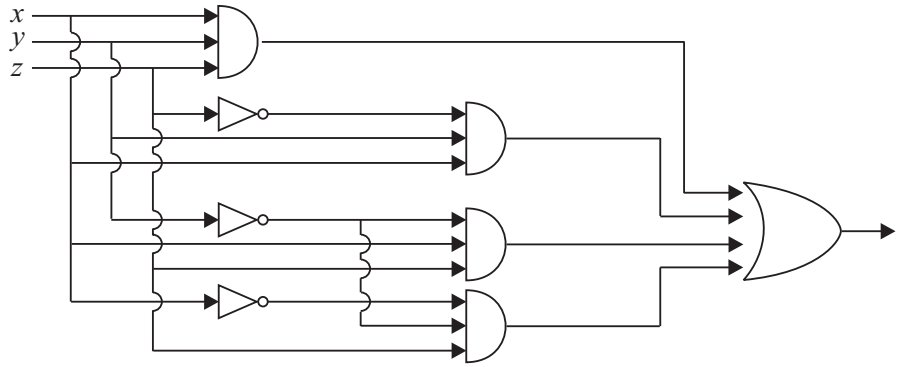


Fig. 7.13.

Solution

The boolean expression represented by the circuit is $f(x, y, z) = xyz + xyz' + xy'z + x'y'z$. Simplify it as much as possible, justifying every step:

$$\begin{aligned} f(x, y, z) &= xyz + xyz' + xy'z + x'y'z \\ &= xy(z + z') + y'z(x + x') \\ &= xy1 + y'z1 \\ &= xy + y'z \end{aligned}$$

Consequently, we can replace the given circuit by the much simpler version in Fig. 7.14.

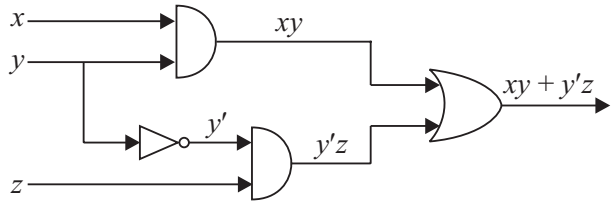


Fig. 7.14. ■

Example 3

Find a simpler combinatorial circuit equivalent to the one represented by the boolean function f (x, y, z) = xyz + xyz' + xy'z + x'yz + xy'z'

Solution

$$\begin{aligned}
f(x, y, z) &= xyz + xyz' + xy'z + x'yz + xy'z' \\
&= xyz + xyz' + xy'z + x'yz + (xy'z' + xy'z') \\
&= xyz + (xy'z + xy'z') + (xyz' + xy'z') + x'yz \\
&= xyz + xy'(z + z') + xz'(y + y') + x'yz \\
&= xyz + xy'1 + xz'1 + x'yz \\
&= xyz + xy' + xz' + x'yz \\
&= xyz + x(y' + z') + x'yz + xyz \\
&= xyz + x(yz)' + yz(x + x') \\
&= x[yz + (yz)'] + yz1 \\
&= x1 + yz \\
&= x + yz
\end{aligned}$$

Therefore, the simpler two-gate circuit in Figure 7.15 can replace the original none-gate one, saving seven gates.

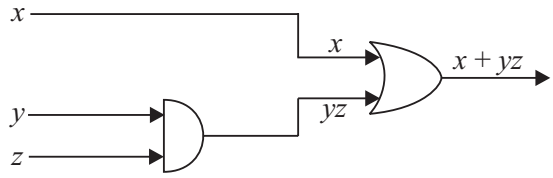


Fig. 7.14. ■

7.5 Karnaugh Map

Maurice Karnaugh developed the graphical method of the Karnaugh map at Bell Laboratories in 1953 to simplify the DNF of boolean expressions. The essence of Karnaugh map lies in grouping minterms that differ by exactly one literal.

e.g. the minterms $xy'z$ and $xy'z'$ differ in exactly one literal and their sum can shrink; $xy'z + xy'z' = xy'(z + z') = xy' \cdot 1 = xy'$.

The k-map has a rectangular grid of square. Each square stands for a possible minterm in the DNF of the boolean expression that represents the circuit. Each contains a 1 if the corresponding minterm exists in the expression.

A 2 × 2 Karnaugh map can help simplify the DNF of a boolean expression in two variables x and y. Since four possible minterms are

feasible –xy, xy', $x'y'$, and $x'y$—the map consists of four squares. Label the squares so the minterms in any two adjacent squares in each row and column differ by exactly one literal; two such squares are **adjacent**. For instance, the squares xy and xy' are adjacent, whereas the squares xy and $x'y'$ are not. The resulting arrangement appears *n* Fig. 7.16.

	y	y'
x	xy	xy'
x'	$x'y$	$x'y'$

Fig. 7.15.

A Karnaugh map for three variables x, y and z expands to a 2 × 4 rectangular grid. Each of the eight squares in Fig. 7.17 corresponds to a possible minterms in x, y, and z. Again, two squares are **adjacent** if the corresponding minterms differ by exactly one literal. For instance, the squares $xy'z$ and xyz are adjacent. To see this geometrically, cut out the grid, bend, and glue the two shorter edges to form the cylinder in Figure 7.17. Two adjacent squares share a boundary.

	yz	yz'	$y'z'$	$y'z$
x	xyz	xyz'	$xy'z'$	$xy'z$
x'	$x'yz$	$x'yz'$	$x'y'z'$	$x'y'z$

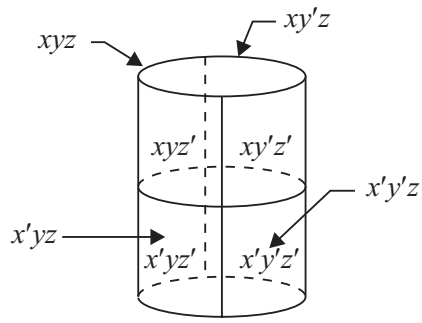


Fig. 7.16.

To simplify a sum of minterms in x, y and z, identify blocks of minterms that can be combined by drawing loops around them. Always begin with the largest block and use the minimum number of blocks. This time a block may be *a* 1 × 1, 1 × 2, 1 × 4, 2 × 2, or a 2 × 4 rectangle. An example of each possibility appears in Figure 7.17.

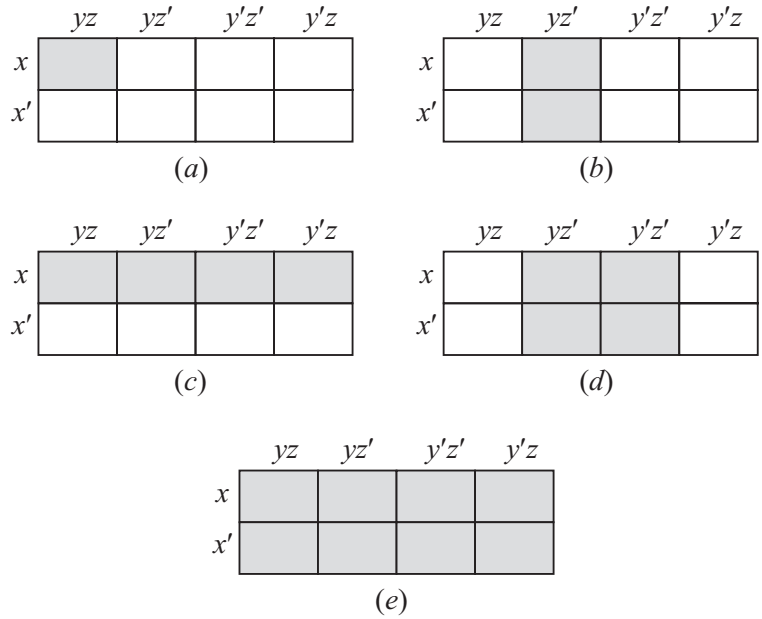


Fig. 2.18.

Notice that the sum of the minterms in the block in Figure 7.17(*d*) be simplified as follows:

$$\begin{aligned} xyz' + xy'z + x'yz' + x'y'z' &= yz'\,(x + x') + y'z'(x + x') \\ &= yz'1 + y'z'1 \\ &= yz' + y'z' \\ &= (y + y')z' = 1z' = z' \end{aligned}$$

Finally, a Karnaugh map for four variables w, x, y, and z encompasses the 4 × 4 grid of Fig. 7.19. Each of the 16 squares represents a possible minterm in w, x, y and z. Geometrically, the grid can be cut, bent and glued to form the doughnut-shaped surface called a **torus.** Two squares are adjacent if they share a border on the torus. The cells $wxy'z$ and $wyxz$ are adjacent; $wxyz$ and $wx'yz'$ are *not*.

	yz	yz'	$y'z'$	$y'z$
wx	$wxyz$	$wxyz'$	$wxy'z'$	$wxy'z$
wx'	$wx'yz$	$wx'yz'$	$wx'y'z'$	$wx'y'z$
$w'x'$	$w'x'yz$	$w'x'yz'$	$w'x'y'z'$	$w'x'y'z$
$w'x$	$w'xyz$	$w'xyz'$	$w'xy'z'$	$w'xy'z$

Fig. 7.19. K. Map of $xy + x'y + x'y'$.

As was done with two- and three-variable expressions, place a 1 in the square corresponding to each minterm in the boolean expression and loop off the 1's into blocks of minterms that can be combined, always looking for the largest block first and using as few blocks as possible. The blocks useful for minimization are 1×1, 1×2, 1×4, 2×2, 2×4, and 4×4. Four such blocks are highlighted in Figure 7.20

For example, the sum of the minterms in Figure 7.19(a) can be simplified:

$$\begin{aligned} wxyz + wxy'z + w'xyz + w'xyz &= wxz(y + y') + w'xz(y + y') \\ &= wxz + w'xz \\ &= (w + w')xz \\ &= xz \end{aligned}$$

We conclude this section with an example that illustrates how to simplify four-variable boolean expressions using a Karnaugh map.

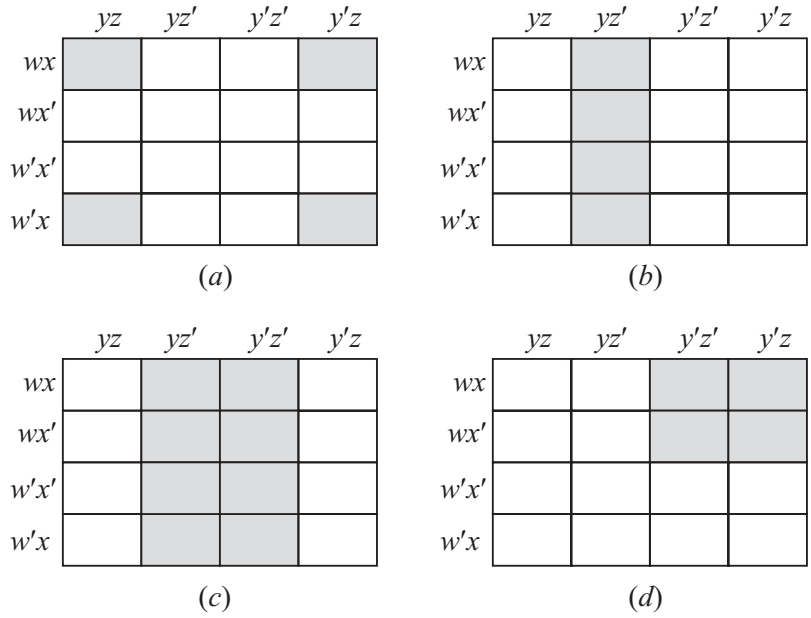


Fig. 7.20.

7.5.1 Don't Care Conditions

Sometimes, combinational circuits may occur that do not accept certain combinations of input signals. Francly speaking, a boolean function f can be generated for which sometime some value combinations never

occur as input. Consequently, we do not care about corresponding output values of f in such cases; the remain unspecified. Such functions are incompletely specified, and these input values are called don't care conditions. To simplify the sum of minterms of such a combinatorial circuit with a Karnaugh map, place a d in each cell corresponding to a don't care condition. This simply means that the corresponding value of the function may be arbitrarily assigned; the minterm may or may not enter the simplification process. Whenever the term can help minimization, its d counts as a 1 in the otherwise standard procedure.

In accomodating through d-cells, the don't care condition of incompletely specified functions in Karnaugh maps emerge as strong workhorses in the reach of combinatorial circuits that mimic the boolean expressions.

7.5.2 Minimization Process

Definition 1: Given two equivalent expressions f and g, we shall say that f is simpler than g if $v_f \leq v_g$, $t_i \leq t_g$, and at least one of these inequlities is strict.

An expression f is said to be minimal if there does not exist an equivalent expression g which is simpler than f.

Note that there may be more than one expressions which are minimal and which are equivalent to any given expression. Minimization in respect of v_f corresponds to having least number of input terminals, whereas that with respect to t_i corresponds to having least number of OR-gates. We neglect the cost of inversion or complementation because flip-flops make available a variable as well as its complement.

Definition 2: Given product terms p and q (*i.e.,* meets of variables or their complements), we say that p is a **subproduct** of q if every x_1 or $\bar{x}_1$ appearing in p also appears in q. We also say that q subsums p in this case. Order of the appearance of $x_1 s$ or $x_1' x$ or the fact that they are juxtaposed or not, is materials because product is commutative (*e.g.,* $x_1 \bar{x}_2$ is a sub-product of $\bar{x}_1 x_2 \bar{x}_3$ or of $x_1 \bar{x}_2 \bar{x}_4$.

Definition 3: Given expression f and g, we say that f implies g if for any assignment $(b_1, \ldots b_n) \in B_2^n$ to the variables for which $f(b_1, \ldots, b_n) = 1$. We also have $g(b_1, \ldots, b_n) = 1$. We also say that f is an **implicant** of g.

For example any product term p in a sum of product form f of an expression is an implicant because if $p(b_1, \ldots, b_n) = 1$ then $f(b_1, \ldots, b_n) = 1$

as well, irrespective of the values of the other product terms (remember that $1 + b = 1$ for $b = 0$ or 1).

Definition 4: A product term p is said to be a prime implicant of an expression f if it implies g but no proper sub-product (*i.e.,* product obtained by deleting x_1 or $\bar{x}_1$ from p) implies f.

For example, $x_1 \bar{x}_2 x_3$ is an implicant of $f(x_1, x_2, x_3) = x_1 \bar{x}_3 x_3 + x_1 x_2 x_3 + \bar{x}_1 x_2 x_3$ because if $x_1 x_2 x_3$ acquires the value 1 for some assignment of binary values to x_1, x_2, x_3, then so does f. But it is not a prime implicant because even its subproduct $x_1 x_3$ is an implicant of f. This can be seen by observing that $x_1 x_3$ has the value 1 whenever $x_1 = 1$, $x_3 = 1$ whatever be x_2.

But in such cases $f(x_1, x_2, x_3) = f(1, x_2, 1)$

$$= 1 \bar{x}_2 1 + 1 x_2 1 + 0 x_2 1 = \bar{x}_2 + x_2 + 0 = 1$$

also *i.e.,* $x_1 x_3$ is an implicant. It is however a prime implicant, for its proper subproducts x_1 and x_3 do not imply f, because we have $f(1, x_2, 0) = 1 \bar{x}_2 0 + 1 x_2 0 + 0 x_2 0 = 0 \; \forall \; x_2$ even while $x_1 = 1$ and $f(0, 0, 1) = 0.1.1 + 0.0.1 + 1.0.1 = 0$ even while $x_3 = 1$.

SOLVED EXAMPLES

Example 1

Find the Karnough map for each boolean expression.

(i) $xy + x'y + x'y'$ *(ii)* $xy' + x'y$.

Solution

Placing *a* 1 in the square corresponding to each minterm and leaving the other square blank, as shown in Fig. 7.21 and 7.22 respectively.

	y	y'
x	1	
x'	1	1

Fig. 7.21. Karnaugh map for $xy + x'y + x'y'$.

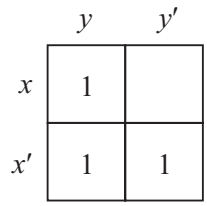


Fig. 7.22 Karnaugh map for $xy' + x'y$. ■

Example 2

Using Karnaugh map, simplify the boolean expression in example 1 above if possible.

Solution

(*i*) First of all, loop off adjacent squares containing 1's. There are two such blocks. (as shown in Fig. 7.23)

$$\begin{aligned} xy + x'y + x'y' &= (xy + x'y) + (x'y + x'y') \\ &= (x + x')y + x'(y + y') \\ &= 1y + x'1. \\ &= y + x' = x' + y. \end{aligned}$$

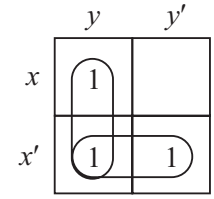


Fig. 7.23.

(*ii*) The squares corresponding to the minterms xy' and $x'y$ are not adjacent, so each forms a block by itself (as shown in Fig. 7.24). Consiquently, the expression $xy' + x'y$ can not be simplified.

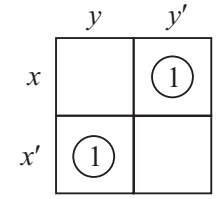


Fig. 7.24.

■

Example 3

Simplify each boolean expression with Karnaugh map.

(i) $E_1 = xyz + xyz' + xy'z + x'y'z$

(ii) $E_2 = xyz + xyz' + xy'z + x'yz + xy'z'$.

Solution

(*i*)

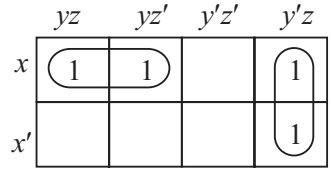


Fig. 7.25

From Fig. 7.26

$$\begin{aligned} E_1 &= (xyz + xyz') + (xy'z + x'y'z) \\ &= xy(z + z') + y'z(x + x') \\ &= xy1 + y'z1. \\ &= xy + y'z. \end{aligned}$$

(*ii*)

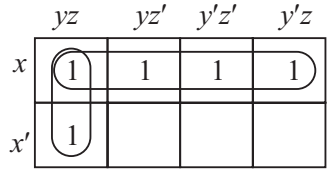


Fig. 7.26

From Fig. 7.26, we have

Since a 1 occurs in both loops, count the corresponding minterm *xyz* twice using the idempotent law:

$$\begin{aligned} E_2 &= (xyz + xyz' + xy'z' + xy'z) + (xyz + x'yz) \\ &= x[y(z + z') + y'(z' + z)] + (x + x')yz \\ &= x(y1 + y1) + 1yz \\ &= x + yz \end{aligned}$$

■

Example 4

Using a Karnaugh map, simplify each boolean expression.

1. $E_1 = wxyz + wxyz' + wxy'z' + wxy'z + w'xy'z$

2. $E_2 = wxy'z + wx'y'z + wx'y'z + w'x'yz + w'x'yz' + w'xyz + w'xyz'$

Solution

1. Place a 1 in the square for each minterm in the expression. Loop off each block, beginning with the largest. The Karnaugh map

is shown in Figure 7.27. Accordingly, the expression E_1 can be pruned:

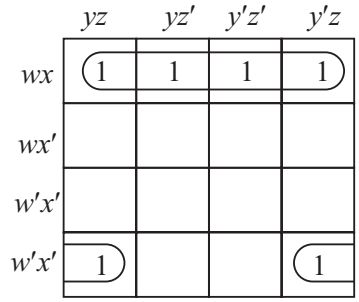


Fig. 7.27

$$\begin{aligned} E_1 &= (wxyz + wxyz' + wxy'z' + wxy'z) + (w'xyz + w'xy'z) \\ &= wx[y(z + z') + y'(z' + z)] + w'xz(y + y') \\ &= wx(y + y') + w'xz \\ &= wx + w'xz \end{aligned}$$

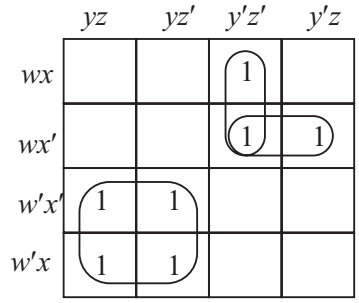


Fig. 7.28

2 Using the Karnaugh map in Figure 7.28, we have:

$$\begin{aligned} E_2 &= (wxy'z' + wx'y'z') + (wx'y'z' + wx'y'z) \\ &\quad + (w'x'yz + w'x'yz' + w'xyz + w'xyz') \\ &= wy'z'(x + x') + wx'y'(z + z') + [w'x'y(z + z') + w'xy(z + z')] \\ &= wy'z' + wx'y' + wx'y' + (w'x'y + w'xy) \\ &= wy'z' + wx'y' + w'y(x + x') \\ &= wy'z' + wx'y' + w'y \end{aligned}$$

■

Example 5

Simplify the boolean expression E represented by the K-map given in Fig. 7.29.

	yz	yz'	$y'z'$	$y'z$
wx	d	d	d	d
wx'	d	d		
$w'x'$	1	1		
$w'x$	1			1

Fig. 7.29

Solution

We draw a loop around each block as usual. Since the top row contains no 1's, exclude it is the minimization process, we can think of its cells as blanks. The resulting blocks are shown in Fig. 7.30, we can verify that

	yz	yz'	$y'z'$	$y'z$
wx	d	d	d	d
wx'	d	d		
$w'x'$	1	1		
$w'x$	1			1

Fig. 7.30

$$E = (wx'yz + wx'yz' + w'x'yz + w'x'yz') + (w'xyz + w'xy'z)$$
$$= x'y + w'xz.$$

Note. If the top row count as a block $E = wx + x'y + w'xz$

Thus, $wx + x'y + wxz = x'y + w'xz$ for all combined values of the variables for which the expression E is specified. Hence the solution depends on a judicious choice of d's. ■

Example 6. *[Binary coded decimal (BCD) expansion]*

Decimal digit	*Binary representation*
0	*0000*
1	*0001*
2	*0010*
3	*0011*
4	*0100*
5	*0101*
6	*0110*
7	*0111*
8	*1000*
9	*1001*

Fig. 7.31

Decimal number	*Binary representation*
10	*1010*
11	*1011*
12	*1100*
13	*1101*
14	*1110*
15	*1111*

Fig. 7.32

Using the above figures (tables) find the blocks of k-map.

Solution

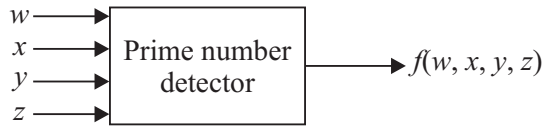


Fig. 7.33

Decimal value	BCD input w	x	y	z	Output
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	d
11	1	0	1	1	d
12	1	1	0	0	d
13	1	1	0	1	d
14	1	1	1	0	d
15	1	1	1	1	d

} don't care conditions

Fig. 7.34

don't care conditions, signaled by the d's in the last column. (Recall that d-cells crop up in the map only if helpful in minimization.) The blocks are shown in Figure 7.35. It follows that:

$$\begin{aligned} f(w, x, y, z) &= (wxyz + wxy'z) + (wx'yz + wx'yz' + w'x'yz \\ &\quad + w'x'yz') + (w'xyz + w'xy'z) \\ &= wxz\,(y + y') + wx'y(z + z') + w'x'y(z + z') \\ &\quad + w'xy(y + y') \\ &= wxz + wx'y + w'x'y + w'xz \\ &= xz(w + w') + x'y(w + w') \\ &= xz + x'y \end{aligned}$$

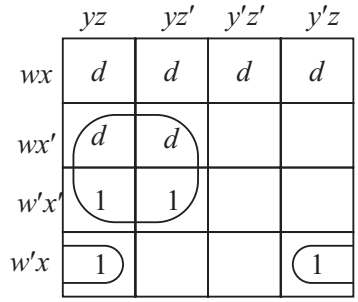


Fig. 7.35

Figure 7.36 gives the logic circuit

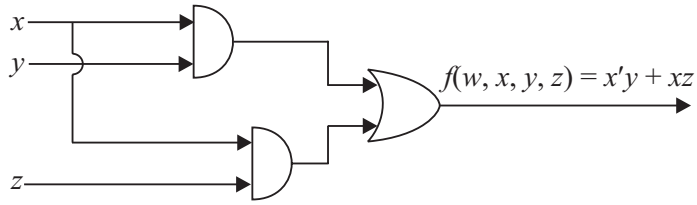


Fig. 7.36

Example 7 *(Digital Display)*

In most of the digital devices such as calculator, microwave ovens, VCD, VCR a maximum of 7 line segments labeled a through g *(as shown in Fig. 7.37). Fig. 7.38 provides the display strategy; the digit 0 lights up the segments a, b, c, d, e, f, 6 lights up c, d, e, f, g and so on*

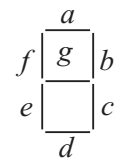


Fig. 7.37

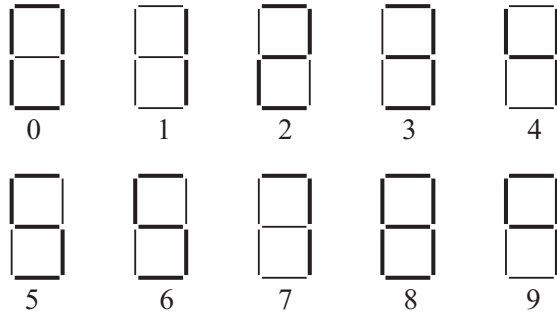


Fig. 7.38

	BCD input				***Inputs***						
Decimal digit	***w***	***x***	***y***	***z***	***a***	***b***	***c***	***d***	***e***	***f***	***g***
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	0	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	0	1	1

Fig. 7.39

DNF, $a = w'x'y'z' + w'x'yz' + w'x'yz + w'xy'z + w'xyz + wx'y'z' + wx'y'z$, which leads to the Karnaugh map in Figure 7.40. The cartographical conclusion, $a = w + x'y + x'y'z' + w'xz$, yields the desired circuit in Fig. 7.41.

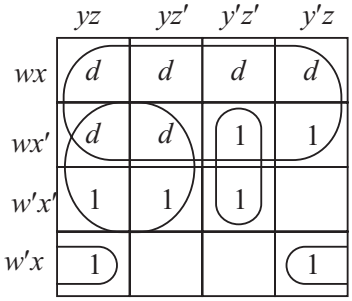


Fig. 7.40

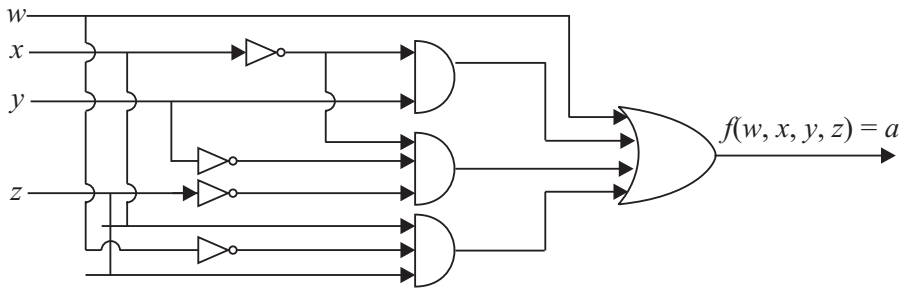


Fig. 7.41

7.6 Finite State Machines

Definition. A sequential machine or a finite state machine is a system $M = (Q, \Sigma, P, \delta, 1)$ where $Q, \Sigma P$ are finite sets of states, input symbols and output symbols. We require that $Q \cap \Sigma\, Q \cap P = f$. Here

$$\delta: Q \times \Sigma \to Q$$

and

$$\lambda: Q \times \Sigma \to P$$

are functions called the next state and output functions respectively. The machine is supposed initially to be in a state q_0.

As the block diagram shows, the reading head of the machine reads symbols, sequentially from an input tape which is divided into equal squares containing one symbol each. Depending upon the current state q and the symbol read, its writing head, writes an output symbol on the output tape, and updates its state to the next. In this new state (which may

be the old one) it is ready to read the next symbol from the input tape. If it reads a symbol a_j from the input tape while in a state q_i, then the output is given by $l(q_i, a_i) \in P$ and the next state is $\delta(q_i, a_i) \in P$ and the next state is $\delta(q_i, a_i) \in Q$. For the case of the serial binary adder considered in the preceding section.

$Q = \{s_0, s_1\}$ $\Sigma = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$ $P = \{0, 1\}$ and the functions δ and λ are given by

$$\delta[s_0, (0, 0) = \delta[s_0, (0, 1)] = \delta[s_0, (1, 0)] = s_0, \delta[s_0, (1, 1)] = s_1$$

so, $\delta[s_1, (0, 0)] = s, \delta[s_1, (0, 1)] = \delta[s_1, (1, 0)]\ \delta[s_1, (1, 1)] = s_1$

and $\lambda[s_0, (0, 0)] = \lambda[s_0, (1, 1)] = 1, \lambda[s_0, (0, 1)] = \lambda[s_0, (1, 0)] = 0$

$$\lambda[s_0, (0, 0)] = \lambda[s_1, (1, 1)] = 1, \lambda[s_1, (0, 1)] = \lambda[s_1, (1, 0)] = 0$$

The machine is started in state s_0.

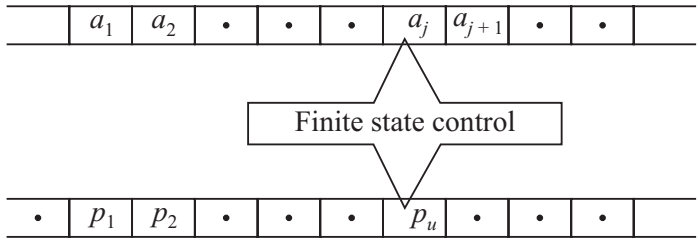


Fig. 7.42.

It is convenient for visual purposes, to depict the working of such a machine by means of a directed graph with lebelled directed edges. Such a graph is called the *transition diagram* of the machine. The nodes are each labelled by the states. From each node, and edge emanates corresponding to each input symbol. Thus given any state q_i and input symbol a_i, if $\delta(q_i, a_j) = q_k$ and $\delta(q_i, a_j) = p_m$, then there is a directed edge from the node labelled by the state q_i to that labelled by q_k. This edge is lebelled by the input a_j and output p_m.

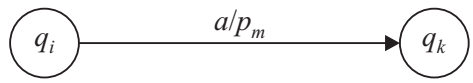


Fig. 7.43

The transition diagram contains the sum-total of all the functional information about the machine in the sense that the sets Q, Σ the maps δ, λ and the subset of P occurring as λ-images (which is all that is needed)

may be recovered from this diagram. For example the transition diagram of the serial binary adder is given as:

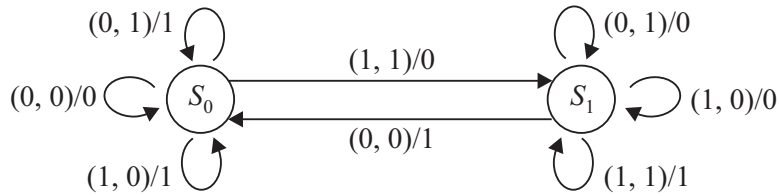


Fig. 7.44. Transition diagram of a serial binary adder.

now what do we do with such a machine? The machine just not only produces an output symbol corresponding to an input symbol. In fact if a finite sequence or string of input symbols is given to it, it processes them one by one to produce a finite sequence of output symbols.

For any set A, we denote by A^* the set of all finite strings of elements of A.

$$A^* = \{a_1 \ldots a_n : a_i \in A, n \geq 0\}.$$

here the members of a string may be distinct or repeated and its length may be any non-negative integer. It is supposed to contain an empty sring $\in$ not containing any symbol and which has length 0. Now A^* contains A in the form of strings of length 1; $\{\in\} \subseteq A^*$, $A \subseteq A^*$.

We define powers of A as the following subsets of A^*.

$$A^2 = \{a_i\, a_j : a_i, a_j \in a_j \in A\} \subseteq A^*$$
$$A^3 = \{a_i\, a_j\, a_k : a_i, a_j, a_k \in A\}$$

as sets of strings of length 2, 3 ...

In this notation

$$A^* = \bigcup_{n=0}^{\infty} A^n$$

where $A_0 = \{\in_0\}$, and $A_1 = A$. The set A^* is a semigroup under the operation of juxtaposition (called the 'catenation operation') of strings and the empty string behaves as the identity element of this semigroup. This semigroup is infinite if $A \neq \phi$ and cancellation laws hold here, although it is not a group. This semigroup A^* is called the free semigroup or free monoid over the alphabet A.

We also note that $A^* = \bigcup_{n=0}^{\infty} A^n = A^* - \{\in\}$ is also a semigroup. We now extend the actions and of input symbols on the states to those of input strings as follows: We take $\delta_0 (q, \in) = q$ and $\delta_1 = \delta$.

Suppose that $a_i \in \Sigma$, for $i = 1, 2, \ldots$ (need not be all distinct). Let $q_1 \in Q$. We denote $\delta(q_1, a_1)$ by q_2, $\delta (q_2, a_2)$ by q_3 etc. so that $(q_1, a_i) = q_{i+1}$. We then define $\delta_2(q_1, a_1, a_2) = \delta[\delta(q_1, a_1), a_2] = \delta(q_2, a_2) = q_3$ in this way we have $\delta_1(q_1, a_1, a_2, \ldots a_1) = q_{i+1}$.

Observe that $\delta_i : Q\ x\ \Sigma^i \to Q$. Since Σn is a disjoint union of the α_s^i these $\delta_i s$ together define a map

$\delta^* : Q\ x\ \Sigma \to Q$ by

$\delta^* (q, x) = \delta_m (q, x)$ since any $x \in \Sigma^*$ belongs to a unique Σ^i *viz.* Σ^m where m = length of x.

In a similar manner if

$\lambda(q_1, a_1) = p_1$, $\lambda(q_2, a_2) = q_2$, ... $\lambda(q_i, a_i) = p_i$ then we get maps $\lambda_1 : Qx\ \Sigma^i \to P$, given by

$$\lambda_1 = \lambda_1\ \lambda_2\ (q_1, a_1, a_2) = \lambda\ (q_2, a_2) = p_2,$$

$\lambda_3(q_1, a_1, a_2, a_3) = \lambda(q_3, a_3) = p_3$

$\vdots$

$\vdots$

$\lambda_i\ (q_i, a_1 \ldots a_n) = \lambda\ (q_i, a_i) = p_i$

Again these maps together define $\lambda^* : Q\ x\ \Sigma + \to P$ given by $\lambda^*\ (q, x) = \lambda^m\ (q, x)$ where m = length of x.

SUMMARY

1. **Boolean Algebra**
   - A **boolean algebra** $\langle B, +, ., ', 0, 1\rangle$ comprises a set B of at least two distinct elements 0 and 1, two binary operators + and ., and a unary operator $'$ that satisfies the commutative, associative, distributive, identity, and complement laws.
   - The zero element 0 and the unit element 1 of a boolean algebra are unique.
   - Every element x of a boolean algebra has a unique complement x'.

- Two important laws of boolean algebra are:

  Absorption laws: $x + xy = x, \quad x(x + y) = x$

  DeMorgan's laws: $(x + y)' = (xy)' = x' + y'$

2. Boolean Function

- A **boolean variable** takes the value 0 or 1.
- A **boolean function** f is a function $f : B^n \to B$.
- Boolean functions can be defined by logic tables or boolean expressions.

3. Boolean Expression

- A **boolean expression** over the boolean variables $x_1, x_2, ..., x_n$ is defined recursively:

  $0, 1, x_1, x_2, ..., x_n$ are boolean expressions.

  If E_1 and E_2 are boolean expressions, so are (E_1), E_1', E_1E_2, and $E_1 + E_2$.

  Only expressions thus obtained are boolean.
- Two boolean expression are **equal** if they yield the same value for every combination of values of the variables.

4. Disjunctive Normal Form (DNF)

- A **literal** is a boolean variable or its complement.
- A **minterm** in n boolean variables $x_1, x_2, ..., x_n$ is a boolean expression $y_1y_2 ... y_n$, where $y_i = x_i$ or x_i', for every i.
- The DNF of a boolean function is a sum of minterms that do not repeat
- The DNF of a boolean function can be constructed using its logic table and the laws of boolean algebra.

5. Functionally Complete Boolean Operators

- A set of boolean operators is **functionally complete** if every boolean function can be defined in terms of them.
- The sets $\{+, ., '\}$, $\{+, '\}$, $\{., '\}$, $\{\downarrow\}$ and $\{\uparrow\}$ are functionally complete.

6. Logic Gates

- A **gate** is a boolean function from B^n to B.
- An **AND gate** accepts two or more boolean values and outputs their boolean product.

- An **OR gate** accepts two or more boolean values and outputs their boolean sum.
- A **NOT gate** (or an **inverter**) accepts a bollean values and outputs their boolean sum.
- A **NOT gate** (or an **inverter**) accepts a boolean value and outputs its complement.
- A **NAND gate** accepts two boolean values x and y and outputs 0 if and only if $x = y = 1$.
- A **NOR gate** accepts two boolean values x and y and outputs 1 if and only if $x = y = 0$.

7. Combinatorial Circuit

- A **combinatorial circuit** produces a unique boolean value for every combination of boolean input values. It has no memory; it can be built using AND, OR, and NOT gates.
- The output of a combinatorial circuit can be described by a boolean function.

8. Adders

- A **half-adder** circuit adds two bits, yielding their sum and carry bits.
- A **Full-adder** circuit accepts two bits and a carry bit from a previous addition, yielding their sum and carry bits. It consists of two half-adders and an OR gate.

9. Minimization of a Combinatorial Circuit

- Two combinatorial circuits are **equivalent** if the corresponding boolean expressions are equal.
- The laws of boolean algebra and Karnaugh maps can simplify combinatorial circuits.

10. Don't Care Conditions

- A combination of boolean values that is not a valid input into a combinatorial circuit is a **don't care condition.**
- A don't care condition triggers a d in the corresponding cell of a Karnaugh map and counts as 1 if it helps simplify the boolean expression.

EXERCISE

1. Determine the output of the following gates:

(*a*) 1101100101, 0100110111 → OR → *X*

(*b*) 1000110010, 1100010100 → OR → *Y*

(*c*) 1111100000, 1100110011 → OR → *Z*

[**Ans.** (*a*) $x = 1101110111$, (*b*) $y = 1100110110$, (*c*) $z = 1111110011$]

2. Using the boolean algebra D_{70}, verify that

(*a*) $(5')' = 5$ (*b*) $7 \oplus (7 \odot 5) = 7$

(*c*) $5 \odot (5 \oplus 7) = 5$ (*d*) $(5 \oplus 7)' = 5' \odot 7'$

3. Let $U = \{a, b, c\}$. Define a suitable function $f : D_{30} \to P(U)$ that processors the operations.

(*a*) $f(x \oplus y) = f(x) \cup f(y)$ (*b*) $f(x \odot y) = f(x) \wedge f(y)$

(*c*) $f(x') = f(x))'$

4. Verify that $x + (yz)'$ is a boolean expression, where the variables x, y and z are boolean variables.

5. Using Boolean algebra, construct the DNF of the boolean function $f(x, y, z) = x(y + z)$

(**Hint.** $x(y + z) = xy + xz = xy \cdot 1 + xz \cdot 1 = xy(z + z') + xz(y + y')$

$= xyz + xyz' + xyz + xzy'$

$= xyz + xyz' + xyz + xy'z = xyz + xyz' + xy'z.$

$\Rightarrow \quad f(x, y, z) = xyz + xy'z + xyz'.$]

6. Prove that in any lattice with a lower bound, every atom is join irreducible.

[**Hint.** If $0 \neq a = a, va_2$ then $a_1 \leq a$. As a is an atom we have $a_1 = 0$ or $a_1 = a \Rightarrow a_2 = a$ or $a_2 = 0$. We have $a = a_2$ or $a = a_1$ shows that a is join irreducible]

7. State and prove store Representation theorem.

[**Hint.** Take any finite boolean algebra $B \exists a$ set S s.t. B and $P(S)$ are isomorphic as boolean algebras.]

8. Prove that the number of elements in a finite boolean algebra B is 2^n where n is the number of its atoms.

9. Prove that each of variables $x_1, x_2, ..., x_n$ considered as expression can be equivalently written as a join minterms.

[**Hint.** We have

$$x_i = 1 \wedge 1 \wedge \ldots \wedge 1 \wedge x_i \wedge 1 \wedge \ldots \wedge 1$$

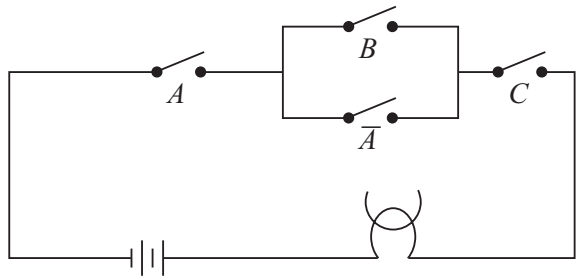
$$= (x_1 \vee x'_1) \wedge (x_2 \vee x'_2) \wedge \ldots \wedge (x_{i-1} \vee x'_{i-1}) \wedge x_i \wedge \ldots \wedge (x_n \vee x'_n)$$

$$= \bigvee_{\alpha_j \neq \alpha_i} x_1^{\alpha_1} \wedge x_2^{\alpha_1} \wedge \ldots x'_j \ldots x_n^{\alpha_1}$$

= A join of all the minterms in which x_i appears as such (*i.e.* uncomplemented).]

10. Prove that every boolean expression is equivalent to a unique expression is disjunctive normal form.

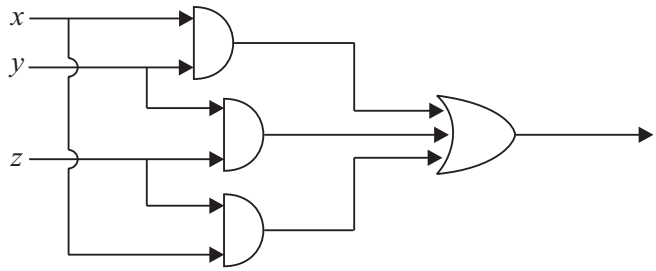
11. Determine the Boolean expression for the given switching circuit.



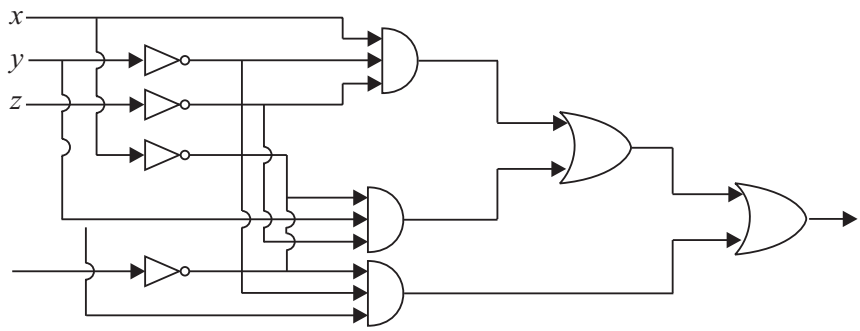
[**Ans.** $(B + (B + \overline{A}) \cdot C)$]

12. Find the output of the following combinatorial circuits.

(*a*)



(*b*)



13. Simplify the boolean expression E represented by a karnaugh map as shown in Fig. below.

	yz	yz'	$y'z'$	$y'z$
wx	d	d	d	d
wx'	d	d		
$w'x'$	1	1		
$w'x$	1			1

[Hint:

	yz	yz'	$y'z'$	$y'z$
wx	d	d	d	d
wx'	d	d		
$w'x'$	1	1		
$w'x$	1			1

14. Using a logic table, verify each statement

(*a*) $(x + y)(y + z) = z + xy$ (*b*) $x + x'yz = x + yz$

15. Evaluate $1 \uparrow (0 \downarrow 1)$ and $0 \downarrow (1 \uparrow 0)$

Suggested Readings

1. J.C. Abbot, *Sets, Lattices and Boolean algebra,* Allyn and Bacon, Boston 1969.

2. Albertson, Hutchinson, *Discrete Mathematics with Algorithms,* Wiley, 1988.

3. B.H Arnold, *Logic and Boolean Algebra,* PH1, 1962.

4. Lovasz, Pelihan, Vesztergombi, *Discrete Mathematics Elementary and Beyond,* Springer, 2006.

5. Thomas Koshy, *Discrete Mathematics and its Applications,* Elsevier 2006.

6. K.H. Roson, *Discrete Mathematics and its Applications,* Mc Graw-Hill, 2008.

7. W.D. Walis, *A biginner's Guide to Discrete to Mathematics,* Birkhauser, 2012.

8. S.K. Yadav, *Discrete Mathematical Structures,* Satya Prakashan, 2001.

❑❑❑

8

Fuzzy Algebra

Lotfi A. Zadeh (b-1921)

Lotfi A. Zadeh (b-1921) was born in Baku, Azerbaijan. He graduated and Postgraduated from University of Tehran and MIT respectively in 1942 and 1946. He received his Ph.D. from Columbia University in 1949 for his thesis on frequency analysis of time-varying networks and began his professional career with the department of Electrical Engineering at Columbia. Later on He joined the Department of Electrical Engineering and Computer Science at the University of California, Berkeley, and served as its chair during 1963-1968. Currently he is a professor and Director of Berkeley Institute of soft computing. His earlier work was on system analysis and Decision analysis in information systems. His current research has been shifted to theory of fuzzy sets and applications in artificial intelligence (AI). His research is now focused on fuzzy logic, soft computing and computing with words.

Dr. Jadeh has authored more than 200 research articles and has been a part of editorial board of more than 50 journals including IEEE. He has received a large numbers of honours and awards throughout the globe.

8.1 Introduction

In our daily life, we always experience a large number of attributes which are not precise. But human brain always processes such imprecise terms. If my doctor asks me "**How are you?**"; I reply "**Almost OK!**" Here the media of communication for information passing is not mathematical but something else which are not well modelled in any way. The hedge "**almost**" is a vague term. But interestingly doctor processed this

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_8

information in his brain and took some action which could be in the form like:

(*i*) Doctor smiles;

(*ii*) Doctor expressed his satisfaction by some movement of his eyes or face;

(*iii*) Doctor thinks of further treatment, if any etc., etc.

Our problems are seen in terms of decision, management, and prediction; solutions are seen in terms of faster access to more information and of increased aid in analyzing, understanding and utilizing the information that is available and is coping with the information that is not. These two elements, a large amount of information coupled with amount of uncertainty, taken together constitute the ground of many of our problems today; complexity. As we are aware of how much we know and of how much we do not know, as information and uncertainty themselves become the focus of our concern, we begin to see our problems as centering around the issue of complexity.

The certain type of data processing or information processing are always done by the world's begest computer "**human brain**". To make this processing faster than human brain, we need to make use of machines. Machine has got no intuition, no intelligence but a large number of circuits and devices. It can be fed with such data which can be processed by it. As a result it can be made intelligent artificially. It can be made expert in any area say: medical diagnosis, chess game, washing of cloths and robot movement etc. For this reason, a mathematical modelling of vague concepts or vague knowledge or imprecesion data or ill defined information is necessary. An important tool, probably one of the most important tools, to do so for such information processing is fuzzy algebra.

As a fact that complexity itself includes both the elements of how much we know, or how well we can describe, and the elements of how much we do not know, or how uncertain we are, can be illustrated with the simple example of driving a vehicle, we can agree that the driving a car is complex. Further more, driving a standard car (transmitted) or stick-shift car is more complex than driving a car with an automatic transmission, one index of which being that more description is needed to cover adequately our think tank of driving in the former case than the later. Because of more knowledge involvement in driving the car of a standard-transmission car, it is more complex. The complexity of driving also involves the degree of our uncertainty. As our uncertainty increases

for instance, in heavy traffic or on unfamiliar roads-so-does he complexity of the tasks. Thus, our preception of complexity increases both when we realize how much we know and when we realize how much we do not know and how we need to know.

8.2 Crisp Sets and Fuzzy Sets

The crisp set is defined as to dichotomize the individuals in some given universe of discurse into two groups *i.e.* members (those certainly being in the set) and nonmembers (certainly do not in the set.)

The fuzzy set introduces vagueness by eleminating the sharp boundary dividing members of the class from nonmembers. A fuzzy set can be defined mathematically by assigning to each possible individual in the universe of discurse a value representing its grade of membership in the fuzzy set. This grade corresponds to the degree to which that individual is similar or compatible with the concept representing by the fuzzy set.

Let us start with a remark that the purpose of a (conventional) set in mathematics is to formally characterize some concept. For instance, the "integer numbers which are greater than or equal to 3 and less than or equal to 10" may be uniquely represented by the following set:

$$\{x \in I: 3 \leq x \leq 10\}$$

i.e. $$\{3, 4, 5, 6, 7, 8, 9, 10\}$$

where I is the set of integers.

However, a serious difficulty occurs if we try to use the above notion of sets to characterize some less precise and somewhat vague concepts, *e.g.,* the **"integers which are more or less equal to 6".**

The source of vagueness is "more or less". It is obvious that the conventional set is somewhat inadequate here; namely, it may be equated with its characteristic function

$$\Psi: X \rightarrow \{0, 1\}$$

which associates with each element of a universe of discourse X (in our case the set of integers I) either 1 or 0 which means that a particular element either belongs to the set or does not, respectively. Thus, there is a clear distinction of the elements belonging and not belonging to the set, or equivalently the transition from 'belonging' to '**not belonging**' to the set is abrupt. But for the **"integers being more or less equal to 6"**, such a precise distinction is evidently artificial. Here it is not possible to adequately fix any precise borderline.

The above line of reasoning suggested by *L.A.* Zadeh in 1965 was to replace the characteristic function Y by the so called **membership function** defined as

$$\mu: X \to \{0, 1\}$$

Thus the transition from 'belonging' to 'not belonging' to a set is now not clear-cut and abrupt, but gradual: from the full membership ($\mu(x) = 1$) through intermediate membership ($0 < \mu(x) < 1$) to the full nonmembership ($\mu(x) = 0$). We have therefore some means for handling vagueness.

Many philosophers and scientists have long since tried to investigate problems pertaining to uncertainty, ambiguity, and vagueness. The only field of mathematical formalism dealing with such questions is the theory of probability. Owing to Zadeh's fruitful attempts, theory of vagueness what Zadeh calls "**fuzziness**" is now sufficiently developed to stimulate researches from a wide range of areas, although a lot of work remains to be done in this framework. Actually fuzzy sets have been suggested for handling the imprecision of real world situations by using truth values between the usual **"true"** and **"false"**. Extensive theories have grown up, exploring the properities of such a set theory, developing various logics for it, and **"fuzzifying"** various branches of pure and applied mathematics, including topology, graph theory, automata, formal languages etc. to name a few. The concept of fuzzy set and membership degree were introduced to provide a possible model for an exact concepts and subjective judgements.

This model was intended to be used in situations when deterministic and/or probabilstic models donot provide a realistic description of the phenomena models under study. Indeed, in a large area of situations, *viz.,* pattern recognition, decision making, large system control, management problems, production scheduling and others, human judgement is often imprecise and the decision maker no more manipulates numbers but handles vague concepts. Most often, on such premises, the decisions themselves are vague and imprecise. Fuzziness does bear a precise meaning. It must be distinguished from generally, ambiguity and error. A symbol is called **"genera"** when it applies to a multiplicity of objects and retains only a common essential feature, it is ambiguous when it may denote several unrelated objects. But the fuzziness of a symbol lies in the lack of precise boundaries of the set of objects to which it

applies. For instance consider the set X of people living in a given town, and subset A or C including only tall people. Obviously some people in X will be definitely tall, others definitely "not tall", but there will be borderline cases. Traditionally, members of the first category are assigned a membership grade 1, those of the second one, a membership grade 0.

More generally, it has been found that the membership functions for unidimensinal concepts vary with the range of the particular context under consideration. Thus, for example, the set of *small* objects can be considered to be a fuzzy subset of the universe of objects. But contrast a *small mouse* with a *small elephant*. Clearly in each instance it is appropriate to talk about a particular item as being a member of the fuzzy subset labelled small, but the range of items that has at one time or another been described as small is so varied that it is unreasonable to assume that a particular item is small in any absolute sense.

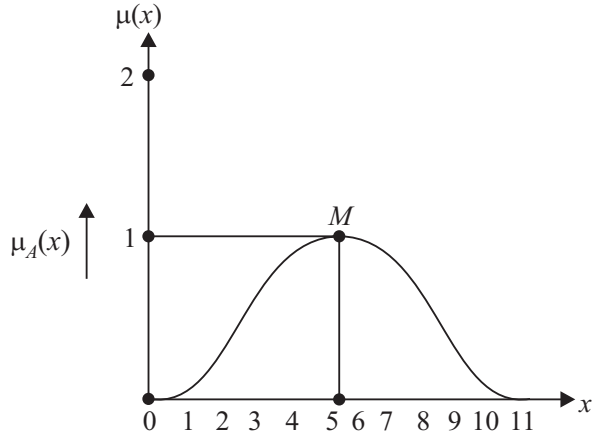


Fig. 8.1: Fuzy set "Integers numbers more or less equal to 6".

Definition: Suppose, X is a non-null set. A fuzzy set A of this set X may be defined by the following set of pairs

$$A = \{(x, \mu_A(x)) : x \in X\}$$

where $$\mu_A : X \to [0, 1]$$

The above mapping is called the membership function of A and $\mu_A(x)$ is the grade of membership or degree of belongingness or degree of membership of $x \in X$ in A.

Thus a fuzzy set is a set of pair consisting of the particular elements of the universe of discurse and their membership grade.

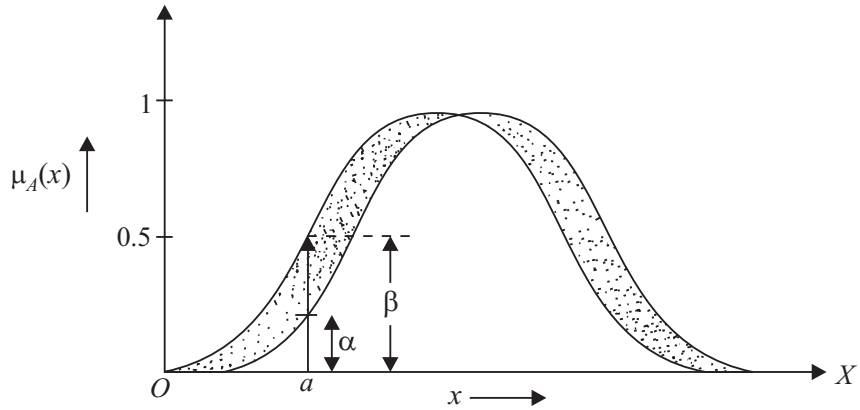


Fig. 8.2: An interval valve fuzzy set ($\mu_A(\alpha) = [\alpha, \beta]$).

8.3 Some Useful Definitions

We have seen that a fuzzy set A of a set $X (\neq \phi)$ is characterized by the membership function

$$\mu_A : X \to [0, 1]$$

Thus for any $x \in X$, the degree of belongingness of it in the fuzzy set A is $\mu_A(x)$, when $0 \leq \mu_A(x) \leq 1$.

If $X = \{x_1, x_2, \ldots x_n\}$, then a fuzzy set A of X could be written as

$$A = \{(x_1, \mu_A(x_1)), (x_2, \mu_2(x_2)) \ldots, (x_n, \mu_A(x_n))\}$$

which sometimes we write in the following way symbolically:

$$A = \left\{ \frac{\mu_A(x_1)}{x_1}, \frac{\mu_A(x_2)}{x_2}, \ldots, \frac{\mu_A(x_n)}{x_n} \right\}$$

(i) **Equality of two fuzzy sets:** Let A and B two fuzzy sets of $X (\neq \phi)$ with membership functions μ_A and μ_B. We say that A and B are equal and written by $A = B$, if and only if

$$\mu_A(x) = \mu_B(x) \quad \forall\, x \in X$$

For example: Suppose $X = \{1, 2, 3\}$ Consider the fuzzy sets A, B, C of X given by

$$A = \left\{ \frac{1}{2}, \frac{2}{7}, \frac{3}{0} \right\}$$

$$B = \left\{ \frac{2}{4}, \frac{1}{2}, \frac{3}{0} \right\}, \text{ and}$$

$$C = \left\{\frac{2}{7}, \frac{3}{0}, \frac{1}{2}\right\}$$

and $A = C, \quad A \neq B, \quad B \neq C$

(*ii*) Containment of fuzzy sets: Let X be a set ($\neq \phi$) and A, B are two fuzzy sets of X with membership function μ_A and μ_B respectively. We say that the fuzzy set A is contained in the fuzzy set B if and only if

$$\mu_A(x) \leq \mu_B(x) \quad \forall\, x \in X$$

We may say, another terminology, that A is a fuzzy subset of B, denoted by $A \subseteq B$.

For example: If X {1, 2, 3} and A, B, C are three fuzzy sets given by

$$A = \left\{\frac{1}{1}, \frac{5}{2}, \frac{1}{3}\right\}, \quad B = \left\{\frac{1}{1}, \frac{4}{2}, \frac{9}{3}\right\}, \text{ and}$$

$$C = \left\{\frac{1}{1}, \frac{6}{2}, \frac{1}{3}\right\}, \text{ then } B \subseteq A,\ C \nsubseteq A$$

(*iii*) Normal Fuzzy set: A fuzzy set of A of set X is called a normal fuzzy set if and only if

$$\max_{x \in X} \mu_A(x) = 1$$

i.e. $\mu_A(x) = 1$ for the least one $x \in X$.

For example: If $x = \{1, 2, 3\}$ and $A = \left\{\frac{1}{2}, \frac{2}{1}, \frac{3}{0}\right\}$ be a fuzzy set of X, then A is a normal fuzzy set.

(*iv*) Support of a fuzzy set: It is the crisp set that contains all elements of X have a non zero membership grade in A.

or

The support of a fuzzy set A of the set X is the classical set

$$\{x \in X : \mu_A(x) > 0\},$$

denoted by support (A).

For example: In the previous example, support (A) = {1, 2}

(*v*) α-cut or α-level set: The α-cut or α-level set of a fuzzy set A of the set X is the following crisp (*i.e.* conventional) set given by

$$A_\alpha = \{x \in X : \mu_A(x) \leq \alpha\}$$

For example: Suppose $X = \{2, 1, 4, 3\}$. Consider the fuzzy set A of X given by

$$A = \left\{\frac{5}{2}, \frac{1}{1}, \frac{1}{4}, \frac{8}{3}\right\}$$

Clearly, $A_1 = \{1, 2, 3, 4\}$

$A_6 = \{3, 4\}$, $A_0 = \{1, 2, 3, 4\}$

$A_1 = \{4\}$, $A_3 = \phi$

Clearly, if $\alpha \geq \beta$, $A_\alpha \subseteq A_\beta$. Also, $A_0 = X$ and $A_\alpha = \phi$, $\forall\ \alpha > 1$.

(*vi*) Infant, Adult, Young and Old Fuzzy sets: The height of a fuzzy set is the largest membership grade attained by any element in that set. A fuzzy set is called normalized when atleast on of its elements attains the maximum possible membership grade. If the membership grades range in the closed interval between 0 and 1, for instance, at least one element must have a membership grade of 1 of the fuzzy set to be considered normalized.

The scalar cardinality of a fuzzy set A defined or a finite universal set X is the summation of the membership grade of all elements of X in A. Thus

$$|A| = \sum_{x \in X} \mu_A(x)$$

Hence scalar cardinality of the fuzzy set infant is 0.

The membership grade of each element of the universal set X in fuzzy set A is less than or equal to its membership grade in fuzzy set B, then A is called a subset of B. Thus. if

$$\mu_A(x) \leq \mu_B(x) \quad \forall\ x \in X$$

Then $A \subseteq B$.

Hence $\mu_{\text{old}}(x) \leq \mu_{\text{adult}}(x)$

8.4 Operations of Fuzzy Sets

The operations among the fuzzy sets are:

(*i*) Complement of a fuzzy set: The complement of a fuzzy set $A \subseteq X$, write A^c is defined as

$$\mu_{A^c}(x) = 1 - \mu_A(x) \quad \forall\ x \in X$$

For example: If $x = \{1, 2, 3\}$ and $A = 0.1/1 + 0.7/2 + 1/3$, then $A^c = 0.9/1 + 0.3/2$.

The idea of complement is even more clear when presented graphically as in Fig. 8.3 given below:

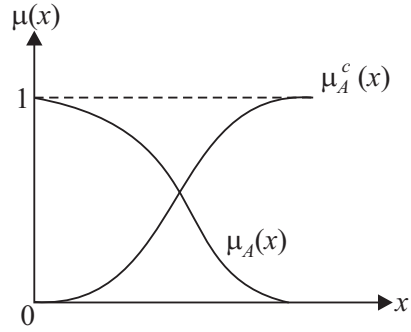


Fig. 8.3: Complement of a fuzzy set

(*ii*) Union of Fuzzy sets: The union of two fuzzy sets $A, B \subseteq X$, written $A \cup B$, is defined as

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x), \quad \forall\, x \in X.$$

where "$\vee$" is the 'maximum' operation.

For example: If $x = \{1, 2, 3, 4\}$,

$$A = \frac{0.2}{1} + \frac{0.5}{2} + \frac{0.8}{3} + \frac{1}{4}$$

$$B = \frac{1}{1} + \frac{0.8}{2} + \frac{0.5}{3} + \frac{0.2}{4} \text{ then}$$

$$A \cup B = \frac{1}{1} + \frac{0.8}{2} + \frac{0.8}{3} + \frac{1}{4}$$

The essence of union may well be seen in Fig. 8.4

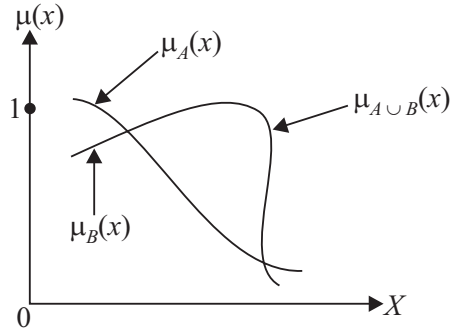


Fig. 8.4: Union of fuzzy sets

(*iii*) **Intersection of Fuzzy sets:** The intersection of two fuzzy sets $A, B \subseteq X$, written $A \cap B$, is defined as $\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x)$, $\forall\ x \in X$, where "$\wedge$" is the 'minimum' operator as shown in Fig. 8.5.

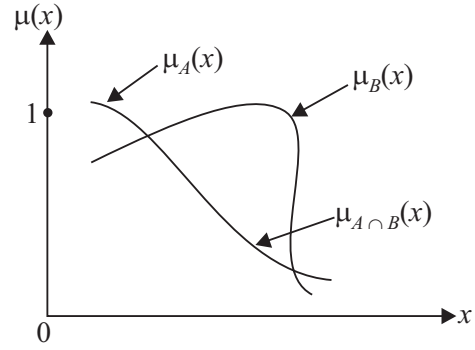


Fig. 8.5: Intersection of fuzzy sets

For example: If $x = \{1, 2\}$, $A = \frac{1}{1} + \frac{0.7}{2}$, $B = \frac{0.5}{1} + \frac{0.9}{2}$ then $A \cap B = \frac{0.5}{1} + \frac{0.7}{2}$.

(*iv*) **Fuzziness and Randomness:** Fuzziness and randomness are some sort of uncertainty, but they are different concepts. The randomness deals with well defined phenomena (events) whose occurrence is uncertain. The fuzziness concerns concepts and entities which are not crisply defined.

And what about a joint occurrence of fuzziness and randomness? Such situations are quite common, *e.g.,* when we ask about the probability of **"good"** weather tomorrow, of **"much lower"** inflation, etc.

(*v*) **Inverse image of a fuzzy sets:** Let X and Y be two sets and f be a mapping from X into Y. Let B be a fuzzy set in Y. Then the *inverse image* $f^{-1}[B]$ of B is the fuzzy set in X with the membership function defined by

$$\mu_{f^{-1}_{[B]}}(x) = \mu_B(f(x)) \quad \forall\ x \in X.$$

(*vi*) **Image of a fuzzy set:** Let A be a fuzzy set in X. Then the image of A, $f[A]$ is the fuzzy set in Y which membership function defined by

$$\mu_{f[A]}(y) = z \in \overset{\sup}{f^{-1}}\ \mu_A(z) \text{ if } f^{-1}(y) \neq \phi$$
$$= 0, \text{ otherwise}$$
$$\forall\ y \in Y, \text{ where } f^{-1}(y) = \{x: f(x) = y\}.$$

Theorem 1: Every fuzzy complement has at most one equilibrium.

Proof: Let c be any arbitrary fuzzy complement.

Hence $c(a) - a = 0 \quad \forall\ a \in [0, 1]$

for any equation, $c(a) - a = b$, $\quad b$ be a real constant assuming that,

a_1, a_2 are two different solutions of the equation $c(a) - a = b$. s.t. $a_1 < a_2$

Since $\qquad c(a_1) - a_1 = c(a_2) - a_2 \qquad \ldots(i)$

because c is monotonic nonincreasing (for all $a, b \in [0, 1]$ then $c(a) \geq c(b)$ *i.e.* c is monotonic nonincreasing)

$\Rightarrow \qquad c(a_1) \geq c(a_2)$

Since $\qquad a_1 < a_2$

$\Rightarrow \qquad c(a_1) - a_1 > c(a_2) - a_2$ a contradiction. ■

Theorem 2: Assume that a given fuzzy complement c has been equilibrium e Then $a \leq c(a)$ iff $a \leq e_c$ and $a \geq c(a)$ iff $a \geq e_c$.

Proof: Assume that $a < e_c$, $\quad a = e_c$ and $a > e_c$ is true.

since c is monotonically nonincreasing,

$c(a) \geq c(e_c)$ for $a < e_c$, $\quad c(a) = c(e_c)$ for $a = e_c$ and $c(a) \leq c(e_c)\ \forall\ a > e_c$

Because $c(e_c) = e_c$ we may rewrite as $c(a) \geq e_c$, $c(a) = e_c$ and $c(a) \leq e_c$ respectively.

from initial assumption,

$c(a) > a, \quad c(a) = a$ and $c(a) < a$ respectively.

Thus $\qquad a \leq e \Rightarrow c(a) \geq a$ and $a \geq e_c$

$\Rightarrow \qquad c(a) \leq a.$

Similarly, the inverse implications can be shown.

Theorem 3: $\lim\limits_{w \to \infty} \min [1, (a^w + b^w)^{1/w}] = \max (a, b)$

Proof: The theorem is obvious whenever.

(*i*) a or b equal to zero *i.e.* $a = 0$ or $b = 0$

(*ii*) $a = b$, because the limit of $2^{1/w}$ as $w \to \infty$ equals 1.

If $a \neq b$ and the non equals $(a^w + b^w)^{1/w}$,

Then

$$\lim_{w \to \infty} (a^w + b^w)^{1/w} = \max(a, b)$$

without loss of generality, we assume that

$$a < b \text{ and } Q = (a^w + b^w)^{1/w}$$

Then

$$\lim_{w \to \infty} = \log Q = \lim_{w \to \infty} \left[\log \frac{(a^w + b^w)}{w} \right]$$

using L' Hospital rule, we get

$$\lim_{w \to \infty} \log Q = \left[\lim_{w \to \infty} \frac{a^w \log a + b^w \log b}{a^w + b^w} \right]$$

$$= \left[\lim_{w \to \infty} \frac{\left(\frac{a}{b}\right)^w \log a + \log^b}{\left(\frac{a}{b}\right)^w + 1} \right] = \log b.$$

Hence $= \lim_{w \to \infty} Q = \lim_{w \to \infty} (a^w + b^w)^{1/w} = b(= \max(a, b))$

It remains to show that the theorem is still valid when the min equals 1 then

$$(a^w + b^w)^{1/w} \geq 1$$

or $a^w + b^w \geq 1$

for all $w \in (0, \infty)$

when $w \to \infty$, the last inequality holds if $a = 1$ or $b = 1$

since $a, b \in [0, 1]$. Hence the theorem is satisfied. ■

Theorem 4: $\lim_{w \to \infty} i_w = \lim_{w \to \infty} [1 - \min\{1, ((1 - a)^w + (1 - b)^w)^{1/w}\}]$

$= \min(a, b)$

Proof: From theorem 3, we know that,

$$\lim_{w \to \infty} \min [1, \{(1 - a)^w + (1 - b)^w\}^{1/w}] = \max [1 - a, 1 - b]$$

Thus, $i_\infty(a, b) = 1 - \max [1 - a, 1 - b]$

Then assume that,

$a \leq b$, then, $1 - a \geq 1 - b$

and $i_\infty(a, b) = 1 - (1 - a) = a$

Hence, $i_\infty(a, b) = \min.(a, b)$ ■

8.5 Interval-Valued Fuzzy Sets (I-V Fuzzy Sets)

The formal fuzzy set representations, especially of verbal expression occurring in a verbal model of a phenomenon, object or process (in a verbal decision procedure or in case of methods of approximate inference) is not often sufficiently adequate. Methods of approximate inference based on fuzzy set theory allow formal representation to be built for verbal decision-making procedures containing vague, fuzzy premises. In experimental investigation, fuzzy inference algorithms almost always assume the form of computer algorithms. That is why both (1) formal, mathematical properties of fuzzy methods of approximate inference and (2) the possibilities of their effective computer representation, belong to important factors determines the final outcome of applying these methods. The former factor determines the 'accuracy' of the representation of a verbal decision procedure by a fuzzy algorithm.

The latter factor determines the accuracy of this algorithm's representation in a computer, computation time and memory loading. Of course, the problem of formulating a verbal decision procedure is a separate issue which goes beyond the frames of fuzzy set theory. As a rule, the membership functions of fuzzy sets representing particular verbal expression can not be defined unequivocally on the basis of available information. *Therefore, it is not always possible for a membership function of the type*

$$\mu: x \rightarrow [0, 1]$$

to assign precisely one point from the interval [0, 1] to each element $X \in X$ *without the loss of a least a part of information.* It should be emphasized, however, that from the point of view of practice, this type of formal representation of a generalized membership function is characterized by some 'redundancy' in relations to reality.

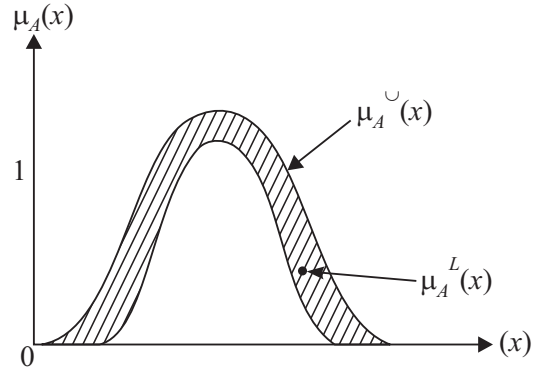


Fig. 8.6: Example of an *I-V* fuzzy set

Definition: An interval-valued fuzzy set (an $i - v$ fuzzy set) A defined on the is given by (Fig. 8.6)

$$A = \left(x\left[\mu_A^L(x), \mu_A^{\cup}\right]\right), x \in X \qquad \ldots(1)$$

where, μ_A^L, $\mu_A^{\cup}$ are fuzzy sets of X such that

$$\mu_A^L, \mu_A^{\cup}$$

Let $\overline{\mu}_A(x)\left[\mu_A^L(x), \mu_A^{\cup}(x)\right], \quad x \in X$

and let $D[0, 1]$ denote the family of all closed intervals contained in the interval $[0, 1]$ *i.e.*

$$\forall\, a_1, a_2 \in [0, 1], a_1 \le a_2, [a_1, a_2] \in D[0, 1]$$

Therefore, for each fixed $x \in X$,

$$\overline{\mu}_A(x) \in D[0, 1]$$

Hence, expression (i) may be rewritten in a more concise form as

$$A = (x, \overline{\mu}_A(x)), \overline{\mu}_A : x \to D\,[0, 1]$$

Thus, interval $\mu_A(x) = \left[\mu_A^L(x), \mu_A^{\cup}(x)\right]$

(a subset of $[0, 1]$ and not a number from the interval $[0, 1]$ is assigned to each element $x \in x$ (Figure). It follows from definition that an arbitrary $i - v$ fuzzy set A is unequivocally represented by the lower and upper limits $\mu_A^{\cup}$ respectively, of its generalized membership function $\overline{\mu}_A$.

8.5.1 Union and Intersection of two I–V Fuzzy Sets

If A and B are two i–v fuzzy sets, then their union is an i–v fuzzy set $A \cup B \in IV\,F\,(X)$ given by

$$A \cup B = \{(x, \overline{\mu}_{A \cup B}(x))\}, \text{ where}$$

$$\mu_{A \cup B}(x) = [\mu^L_{A \cup B}(x), \mu^{\cup}_{A \cup B}(x)], x \in X,$$

where $\mu^L_{A \cup B}(x) = \max\,[\mu_A^L(x), \mu_B^L(x)]$

$$\mu^{\cup}_{A \cup B}(x) = \max\,[\mu_A^{\cup}(x), \mu_B^{\cup}(x)], \forall\, x \in X.$$

This definition may be generalised, in a natural way, to the case of a union of n i–v fuzzy sets.

If A and B are two i–v fuzzy sets, then their intersection is an i–v fuzzy set $A \cap B \in i.v.\ F(x)$ given by (Fig. 8.7)

$$A \cap B = \{(x, \overline{\mu}_{A \cap B}(x))\}$$

$$\overline{\mu}_{A \cap B}(x) = [\mu^L_{A \cap B}(x), \mu^{\cup}_{A \cap B}(x)], x \in X.$$

where $\mu^{L}_{A \cap B}(x) = \min [\mu^{L}_{A \cap B}(x), \mu^{L}_{B}(x)]$,

$\mu^{\cup}_{A \cap B}(x) = \min [\mu^{\cup}_{A}(x), \mu^{\cup}_{B}(x)]$.

This definition also may be generalised.

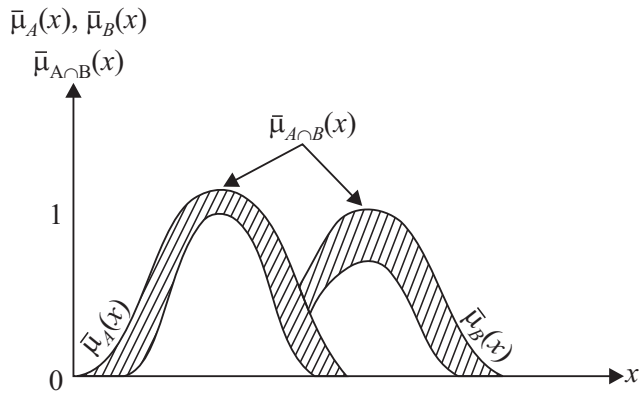


Fig. 8.7: Example of an *i-v* fuzzy set

8.6 Fuzzy Relations

Any relation between two sets X and Y is called a binary relation. It is usually denoted by $R(X, Y)$. When $X \neq Y$, binary relations $R(X, Y)$ are called bipertite graph and when $X = Y$, they are called directed graphs or digraphs.

The symbol R representing the binary relation is used xRy when $(x, y) \in R(X, Y)$. For fuzzy relation we write ${}_{a/x}R_{y}$ when $\mu_R(x, y) = \alpha$.

If $R(X, Y)$ is a fuzzy relation, its domain is the fuzzy set domains as dom $R(X, Y)$ with membership function defined by,

$$\mu_{\text{dom}R}(x) = \max_{y \in Y} \mu_R(x, y). \quad \forall\, x \in X.$$

when $R(X, Y)$ is a fuzzy relation, its range is a fuzzy set ran $R(X, Y)$ with membership function defined by,

$$\mu_{\text{ran}R}(Y) = \max_{x \in X} \mu_R(x, y)\ \forall\, y \in Y.$$

The height of a fuzzy relation R is a number $h(R)$ defined by

$$h(R) = \max_{y \in Y} \cdot \max_{x \in X} \cdot \mu_R(x, y)$$

The largest membership grade attained by any pair (x, y) in R.

If $h(R) = 1$, then the relation is called *normal*; otherwise it is called *subnormal*.

If $\mu_{R\,(X \to Y)}\,(x, y) > 0$, then y is called the *image* of x in R.

If R is a mapping and, at least one of the members of its range appears more than once in R, then the relations is called *many-to-one* and if each element of the range appears exactly once in the mapping, it is called a *one-to-one* relation.

Note: (*i*) The fuzzy relation $R(X, X)$ is *reflexive* iff. $\mu_R\,(x, x) = 1$.

(*ii*) A fuzzy relation is *symmetric* iff $\mu_R\,(x, y) = \mu_R\,(y, x)$.

(*iii*) A fuzzy relation is *transitive* iff $\mu_R\,(x, z) \geq \max\limits_{y \in Y} \min\,[\mu_R\,(x, y), \mu_R\,(y, z)]$.

Definition: Let x, y be two sets A fuzzy relation $\Re$ from X to y is a fuzzy subset of $X \times Y$, characterized by its membership function:

$$\mu_R : X \times Y \to [0, 1]$$

A fuzzy relation $\Re$ from X to Y will be referred to as $\Re\,(X \to Y)$.

For example: Say, $X = \{2, 5\}$, Y = {1, 9, 7}

Here 2 is related to 7 with

Strength = 0.3 etc.

$\Re$

$x \to y$	1	9	7
2	0.1	0	0.3
5	0.7	0.5	0.9

(*i*) Sup-min composition of a relation with a fuzzy set: When A is a fuzzy subset of X, the sup-min composition of $\Re$ and A is fuzzy subset B of Y defined by

$$\mu_B\,(y) \textit{ i.e. } \mu_{R\,o\,A}(y) = \sup_{x \in X}\,\{\min\,(\mu_A\,(x), \mu_R\,(x, y))\}\ \forall\, y \in Y.$$

which is usually rewritten with $\vee$ (sup), and $\wedge$ (min) operations, deleting $x \in X$' when no ambiguity arises, as below:

$$\mu_{R\,o\,A}\,(y) = x\ \{\mu_A\,(x) \wedge \mu_R\,(x, y)\}\ \forall\, y \in Y.$$

(*ii*) Sup-min composition of two fuzzy relations: Let $Q(X \to Y)$ and $R(Y \to Z)$. The sup-min composition $R\ o\ Q$ is a fuzzy relation from x to z, defined by

$$\mu_{R\,o\,Q}(x, z) = y^{v}\{\mu_{Q}(x, y) \wedge \mu_{R}(y, z)\}$$
$$\forall\ (x, z) \in X \times Z, y \in Y$$

For example: If $x = \{1, 2\}$, $Y = \{1, 2, 3\}$ and $Z = \{1, 2, 3, 4\}$, then the composition of two fuzzy relations, R and S, given below in the matrix form is

$$\underset{y}{R} \circ \underset{z}{S} =$$

$$\begin{array}{c} \\ x \\ \\ \end{array}\begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & \end{array} \\ \begin{bmatrix} 0.3 & 0.8 & 0.1 \\ 0.9 & 0.7 & 0.4 \end{bmatrix} \end{array} O\ Y \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0.7 & 0.6 & 0.4 & 0.1 \\ 0.4 & 0.1 & 0.7 & 0.2 \\ 0.5 & 0.9 & 0.6 & 0.8 \end{bmatrix} \end{array} = \begin{array}{c} \\ x \\ \\ \end{array}\begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0.5 & 0.9 & 0.7 & 0.8 \\ 0.7 & 0.7 & 0.7 & 0.4 \end{bmatrix} \end{array}$$

(*iii*) *i–v* Fuzzy Relation: An $i - v$ fuzzy relation $\Re$ is an $i - v$ fuzzy set in a cartesian product $X \times Y$ of universe x and y. The membership value of (x, y) *i.e.* $\overline{\mu}_{R}(x, y)$ in $\Re$ estimates the interval of the strength of the link between x and y.

$$\mu_{R}^{\cup}(x, y) \text{ and } \mu_{A}^{L}(x, y)$$

are respectively called the maximum strength and minimum strength of the link between x and y. Two $i - v$ fuzzy relations $\Re_1$ and $\Re_2$ are equal if

$$\forall\ (x, y) \in X \times Y$$
$$\mu_{R_1}(\xi, \psi) = \mu_{R_1}(x, y).$$

(*iv*) Composition: Let A be an $i - v$ fuzzy set of x and $\Re$ be and $i - v$ fuzzy relation from x and y. Then composition of $\Re$ with A is $B = \Re\ o\ A$, which is an $i - v$ fuzzy set of Y defined by

$$\overline{\mu}_{RoA}(y) = \left[\underset{x \in X}{\text{Infmum}}\ (\mu_{A}^{L}(x) \wedge \mu_{R}^{L}(y, z)),\ \underset{x \in x}{\text{Supremum}}\ (\mu_{A}^{\cup}\, x \wedge \mu_{R}^{\cup}(y, z))\right]$$

Let $\Re$ and $\Im$ be two $i - v$ fuzzy relations on $X \times Y$ and $Y \times Z$ respectively. Then the composite relation $\Re\ o\ \Im$ is an $i - v$ fuzzy set in $X \times z$ with membership function defined by:

$$\overline{\mu}_{RoS}(x, y)$$
$$= \left[\underset{x \in Y}{\text{Inf}}\left(\mu_{S}^{L}(x, y) \wedge m_{R}^{L}(y, z)\right),\ \underset{y \in Y}{\text{Sup}}\left(\mu_{S}^{\cup}(x, y) \wedge \mu_{R}^{\cup}(x, z)\right)\right]$$

Thus an interval valued link between x and z is established through Y. This $\mathcal{R}\ o\ \mathcal{S}$ is known as a composition of $i - v$ fuzzy relations which we shall refer to henceforth as *Civ FR*.

An i–v fuzzy relation $\mathcal{R}$ on $X \times X$ is said to be

(i) Reflexive: iff "$\forall\ x \in X, \overline{\mu}_R(x, x) = [1, 1]$"

(ii) Symmetrical: iff $\forall x_1, x_2 \in X$,

$$\overline{\mu}_R(x_1, x_2) = \overline{\mu}_R(x_2, x_1)$$

(v) Inverse of an $i - v$ fuzzy relation: If $\mathcal{R}$ is an $i - v$ fuzzy relation on $X \times Y$, its inverse $\mathcal{R}^{-1}$is an $i - v$ fuzzy relation on $Y \times X$ such that $\overline{\mu}_{R-1}(y, x) = \overline{\mu}_R(x, y)$.

Clearly, $\mu^L_{R-1}(y, x) = \mu^L_R(x, y)$ and

$$\mu^{\cup}_{R-1}(y, x) = \mu^{\cup}_R(x, y).$$

Proposition

(i) $(\mathcal{R}^{-1})^{-1} = \mathcal{R}$

(ii) $(\mathcal{R}\ o\ \mathcal{S})^{-1} = \mathcal{S}^{-1}\ o\ \mathcal{R}^{-1}$

where $\mathcal{R}$ and $\mathcal{S}$ are any two $i - v$ fuzzy relations on $X \times Y$ and $Y \times Z$ respectively.

(i) $\quad \overline{\mu}_{(R^{-1})^{-1}}(x, y) = \mu_{R^{-1}}(y, z)\, \mu_R(x, y)$

$$\forall\ x \in X, y \in Y.$$

$\Rightarrow \quad (\mathcal{R}^{-1})^{-1} = \mathcal{R}$

(ii) $\quad \mathcal{R} : X \to Y, \quad \mathcal{S} : Y \to Z,$

$\Rightarrow \quad \mathcal{R}\ o\ \mathcal{S}: X \to Z, \quad \mathcal{R}^{-1}: Y \to X,$

and $\quad \mathcal{S}^{-1}: Z \to Y.$

$\Rightarrow \quad (\mathcal{R}\ o\ \mathcal{S})^{-1}: Z \to X, \quad \mathcal{S}^{-1}\ o\ \mathcal{R}^{-1} : Z \to X.$

Now, $\quad \overline{\mu}_{(R \circ S)^{-1}}(z, x) = \overline{\mu}_{(R \circ S)}(x, z)$

$$= \left[\operatorname*{Inf}_y \left(\mu^L_S(x, y) \wedge \mu^L_R(y, z) \right), \operatorname*{sup}_y \left(\mu^{\cup}_S(x, y) \vee \mu^{\cup}_R(y, z) \right) \right]$$

$$= \left[\operatorname*{Inf}_y \left(\mu^L_{R^{-1}}(x, y) \vee \mu^L_{S^{-1}}(y, z) \right), \operatorname*{sum}_y \left(\mu^{\cup}_{R^{-1}}(x, y) \vee \mu^{\cup}_{S^{-1}}(y, z) \right) \right]$$

$$= \mu_{\mathcal{S}^{-1} \circ \mathcal{R}^{-1}}(z, x).$$

8.6 Fuzzy Measures

A fuzzy measure is defined as a function

$$g: \mathfrak{B} \to [0, 1]$$

where $\mathfrak{B} \in P(x)$ is a family of subsets of X s.t.

(*i*) $\phi \in \mathfrak{B}$ and $X \in \mathfrak{B}$

(*ii*) If $A \in \mathfrak{B}$, then $\overline{A} \in \mathfrak{B}$

(*iii*) $\mathfrak{B}$ is closed under the operation of set union, *i.e.*

If $A \in \mathfrak{B}$ and $B \in \mathfrak{B}$. Then $A \cup B \in \mathfrak{B}$.

The set $\mathfrak{B}$ is called a *Borel field* or σ-*field*.

since $A \cup B \supseteq A$ and $A \cup B \supseteq B$ then, we have

$$\max [g(A), g(B)] \leq g(A \cup B) \qquad \ldots(i)$$

since $A \cap B \subseteq A$ and, $A \cap B \subseteq B$, we have

$$g(A \cap B) \leq \min [g(A), g(B)]$$

for fuzzy measure, the following axioms hold:

(*i*) $g(\phi) = 0$ and $g(x) = 1$, (boundary condition)

(*ii*) $\forall\ A, B \in P(x)$, if $A \leq B$, then $g(A) \leq g(B)$ (monotonicity)

(*iii*) For every sequence $(A_i \in P(x): i \in N)$ of subsets of X, if either $A_1 \subseteq A_2 \subseteq \ldots$ or $A_1 \supseteq A_2 \supseteq \ldots$

Then $\lim_{i \to \infty} g(ai) = g(\lim_{i \to \infty} ai)$

8.7.1 Belief and Plausibility Measures

A *belief measure* is a function

$$\text{Bel} : P(x) \to [0, 1]$$

satisfying fuzzy measures.

Let $A \subset B$ and $C = B - A$ than $A \cup C = B$ and $A \cap C = \phi$.

We obtain

$$\text{Bel}\ (A \cup C) = \text{Bel}\ (B) \geq \text{Bel}\ (A)\ + Bd\ (C) - \text{Bel}\ (A \cap C)$$

since $A \cap C = \phi$ and Bel $(\phi) = 0$ then.

Bel $(B) \geq$ Bel (A), Bel $(B) \geq$ Bel (A) + Bel (C)

A *plausibility measure* is a function

$$Pl : P(x) \to [0, 1]$$

satisfies fuzzy measure.

8.7.2 Probability Measure

A probability measure can be explained as

$$\text{Bel}(A \cup C) = \text{Bel}(A) + \text{Bel}(B) \text{ whenever } A \cap B = \phi$$

$$\Rightarrow \quad \text{Bel}(A) = Pl(A) = \sum_{x \in A} p(x)$$

Then $$p(A) = \sum_{x \in A} p(x)$$

Similarly

$$p_x(x) = \sum_{y \in Y} p(x, y), \quad \forall\, x \in X$$

$$p_y(y) = \sum_{x \in X} p(x, y), \quad \forall\, y \in Y$$

So $$p(x, y) = p_x(x) \,.\, p_y(y) \qquad \forall\, x \in X \text{ and } y \in Y.$$

8.7.3 Uncertainty and Measures of Fuzziness

These are six clusters to define uncertainty:

(*i*) Not certainly known: questionable: Problematical.

(*ii*) Vague: not defined definitely or determined.

(*iii*) Doubtful: not having certain knowledge: not sure.

(*iv*) Ambiguous.

(*v*) Not steady or constant: varying.

(*vi*) Liable to change or vary: not dependable or reliable.

Each of two distinct forms of uncertainty: vagueness and ambiguity is connected with a set of kindred concepts. Some of the concepts connected with vagueness are fuzziness, haziness, cloudiness, unclearness, indistinctiveness and one-to-many relations, variety, generality and divergence. The concept of fuzzy measure provides a general framework for

dealing with ambiguity. Three types of ambiguity are readily recognisable within this framework. The first is connected with the size of the subsets that are designated by a fuzzy measure as prospective locations of the element in question. The second type ambiguity is exhibited by disjoint subsets of X that are designated by the given fuzzy measure as prospective locations of the element of concern. The third type of ambiguity is associated with the number of subsets of X that are designated by a fuzzy measure as prospective locations of the element under consideration and that do not overlap or overlap only partially.

Generally, a measure of fuzziness is a function.

$$f : \tilde{P}(X) \longrightarrow \mathbb{R}$$

where $\tilde{P}(X)$ denotes the set of all fuzzy subsets of X. *i.e.* the function f assigns a value $f(A)$ to each fuzzy subset A of X that characterizes the degree of fuzziness of A.

The unique requirement states that the degree of fuzziness must be zero for all crisp sets in $\tilde{P}(X)$ and only for these.

formally,

(*i*) $f(A) = 0$ iff A is a crisp set.

(*ii*) If $A < B$, then $f(A) \leq f(B)$

(*iii*) $f(A)$ assumes the maximum value iff A is maximally fuzzy.

8.7.4 Uncertainty and Information

In human communication system, the concept of information is more complex. It almost invariably involves three basic aspects:

(*i*) **Syntactic:** regarding the relationship among the signs that are employed in the communication;

(*ii*) **Sementic:** regarding the relationship between the signs and the entities which they stand, *i.e.* regarding the designation of the meaning of the signs;

(*iii*) **Pragmatic:** regarding the relationship between the signs and their utilities;

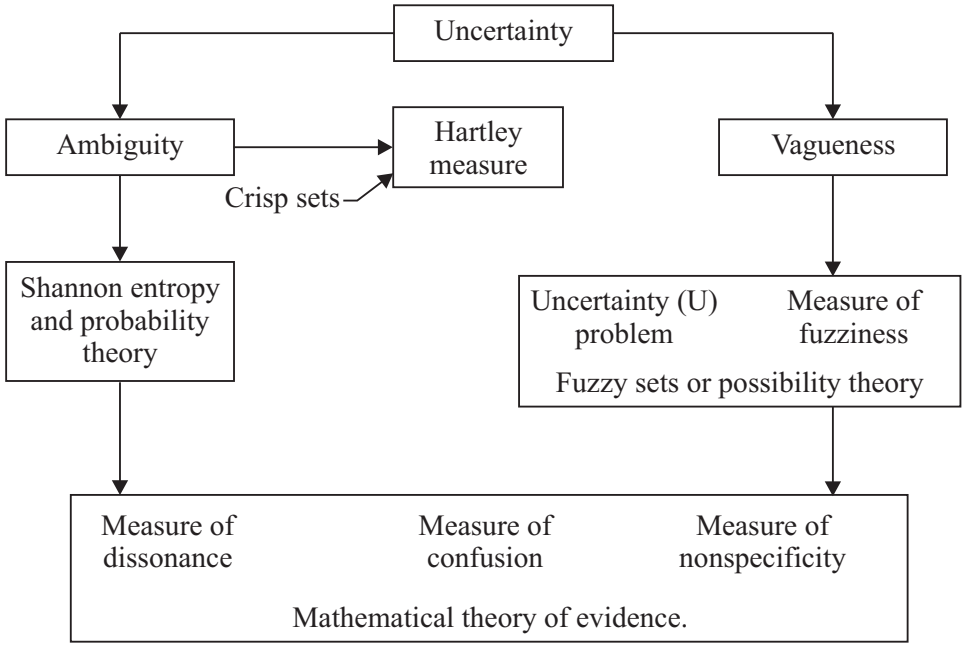


Fig. 8.8: Uncertainty measures in Fuzzy sets

8.8 Applications of Fuzzy Algebras

The successful applications of the mathematical models and tools is extremely diverse and widespread, and the sample included here is by no means exhaustive. Each section of fuzzy algebra includes a brief overview of the applications area followed by some specific illustrative examples drawn from the literature. The following are the main applications of fuzzy algebra in various fields:

8.8.1 Natural, Life and Social Sciences

To understand human system, behaviour and thoughts, fuzzy algebra (especially uncertainty and information) is used. The application of fuzzy set theory include explorations within psychology and cognitive science of concept formation and manipulation, memory and learning, as well as studies in the field of sociology, economics, ecology, metrology, biology and other physical sciences.

The science of meteorology deals with a vastly complex system which, in truth, encompasses the entire planet. For time and season,

meteorological descriptions as well as forecasts have always relied on the kind of robust summary offered by vague linguistic terms such as hot weather, drought, or low pressure. Applications of fuzzy algebra to meteorology constitute attempts to deal with the complexity of the study by taking advantage of the representation of vagueness offered in the mathematical formalism. To understand of climatological changes based on the examination of statistical record of weather pattern collected over period of time and forecast systems.

The process of interpersonal communication consists of a vast array of different types of simultaneously communicated signals (like words, voice tone, body posture, clothing etc.) many of which conflict with each other. It is therefore difficult to determine the precise intention and meaning of the communication, both because of distortion from environmental noise and because of ambivalence on the part of the sender. The receiver must respond appropriately in the face of this fuzzy or vague information.

The receiver of the communication possesses some background information in the form of probabilities or possibilities of the signals which can be expected.

If $p(x_1)$, $p(x_2)$, …, $p(x_n)$ represented the probabilities associated to each of the signals $x_1, x_2, x_3, \ldots, x_n \in X$, then the probability of the fuzzy events of the receipt of message M is given by

$$p(M) = \sum_{x \in X} \mu_M(x) p(x).$$

The total possibility of the fuzzy message M is calculated as,

$$r(M) = \max_{x \in X} [\min (\mu_M(x), r(x)]$$

If r_0 indicates the initial possibilistic expectations subsequent to the receipt of message M, then

$$r_1(x) = \min[r_0^{\alpha}(x), \mu_m(x)] \quad \forall\, x \in X, \alpha \text{ be degree of } M.$$

An additional complication occurs when the receiver may introduce distortion in the message because of inoxsistancy with the expectations.

Let $$s(M, r) = \max_{x \in X} [\min(\mu_M(x)), r(x))]$$

Let M' denote the message which the receiver hears, where

$$\mu_{M'}(x) = \mu_M^s(x) \quad \forall\, x \in X.$$

Then $$r_1(x) = \min \left[r_0^{1-s}(x),\ \mu_{M'}(x) \right] \quad \forall\ x \in X.$$

A fuzzy response set $A \in Y$ can be generated by composing the appropriateness relation R with the fuzzy message M,

$$A = R\ o\ M$$

or $$\mu_A(y) = \max_{x \in X} [\min (\mu_R(y, x), \mu_M(x))] \quad \forall\ y \in Y.$$

8.8.2 Engineering

Although the use of fuzzy algebra in engineering has been loss extensive than the use of information theory, a broad spectrum of applications has occurred in the literature, ranging from civil engineering and archetecture to automatic control and robotics. **Fuzzy** and **control** is the most developed area of application of fuzzy algebra in engineering. (As shown in Fig. 8.9)

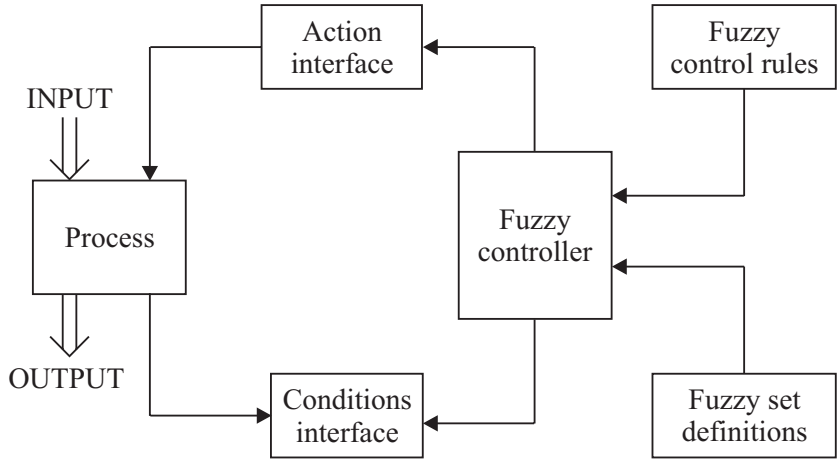


Fig. 8.9: Fuzzy controller

The operation of fuzzy controller consists of four stages. First, a measurement must be taken of the state variables to be monitored during the process. Next, these values must be translated by a component known as a condition interface into fuzzy linguistic terms; these terms are specified by the membership function of the fuzzy sets, which are defined on the appropriate universe of discourse. These linguistic terms are then used in the evaluation of the fuzzy control rules. The result of application of the rules is itself a fuzzy set defined on the universe of possible control actions. At the end, based on this fuzzy set, the component called an

action interface must produce the crisp control action, which is then executed. The design of fuzzy controller, therefore, requires specification of all the fuzzy sets and their membership functions defined for each input and output variables, compilation of an appropriate and complete set of heuristic control rules that operate on there fuzzy sets, and a determination of the method of choosing a crisp output action based on the fuzzy result generated by the control rules. Fuzzy control rules are evaluated by means of the compositional rule of inference. This rule is essentially a fuzzy generalization of the traditional modus porens rule of logical inference. Given statements from '**If A then B**' in which A and B are fuzzy sets defined on universal set X of inputs and Y of outputs, respectively, then the rule base composed of these statements can be fully captured by a fuzzy relation R defined on $X \times Y$. Each statements S is itself represented by a relation specified by

$$\mu_S(x, y) = \min(\mu_A(x), \mu_B(y)), \forall\ x \in X \text{ and } y \in Y.$$

The relations S are then aggregated to form the rule base relation R.

Actual input fuzzy set A' can be used to compute the resulting output fuzzy set B', which is induced on Y by

$$B' = A' \ o \ R$$

The max-min composition is used here such that

$$\mu_{B'}(y) = \max_{x \in X} [\min(\mu_{A'}(x), \mu_R(x, y))]$$

Automatic control of traffic signals based on a stochastic model has long been in use for many cities. A fuzzy logic controller (FLC) was designed and tested by Mamdani and Pappis in 1977 for a single intersection of two one way streets. The fuzzy control rules employed have the form

IF T = *medium*
AND A = *mt (medium)*
AND Q = *lt (small)*
THEN E = *medium*

ELSE

IF T = *long*
AND A = *mt (many)*
AND Q = *lt (medium)*
THEN E = *long*

The input variables T, A, and Q represent the time since the last change, the number of arrivals through the green light, and the number of vehicles waiting in a queue at the opposing red light, respectively.

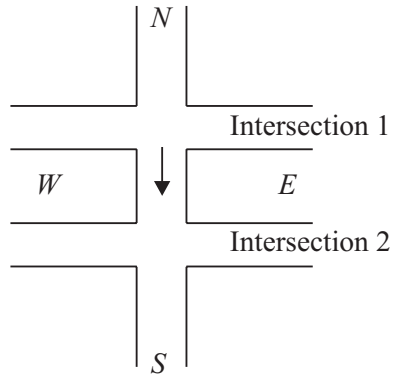


Fig. 8.10: Two Traffic intersections under fuzzy control

The take off and take over of an aircraft involve maneuvering and maintaining flight is an appropriate course to the airport and then along the optimum "glide path" trajectory to touchdown on the runway. This path is usually provided to the pilot by an instrument landing system (ILS), which transmits two radio signals to the aircraft as a navigational aid. These orthogonal radio beams are known as the localizer and glide slope and are transmitted from the ends of the runway in order to provide the approaching aircraft with the correct trajectory for landing. The pilot executing such a landing must monitor cockpit instruments that display the position of the aircraft relative to the desired flight path and make appropriate correction to the controls. All the inputs and outputs are described with the use of five fuzzy sets; *negative big (NB)*, *negative medium (NM)*, *insignificant change (IC)*, *positive medium (PM)* and *positive big (PB)*. These rules were elicited from and experienced pilot and have the form

$$\begin{aligned} \textit{IF rate of descent} &= PM \\ \textit{air speed} &= NB \\ \textit{glide slope} &= PB \end{aligned}$$

THEN

$$\begin{aligned} \textit{rpm change} &= PM \\ \textit{elevator angle change} &= IC. \end{aligned}$$

Two different methods are employed for selecting crisp control actions from the fuzzy directions provided by control rules. One of these calculates the center of gravity g of the fuzzy set C indicating the control action as under

$$g = \sum_{x \in X} \frac{x \cdot \mu_c(x)}{\sum_{x \in X} \mu_c(x)}$$

where X is the appropriate universal set of output actions, each represented by a control change magnitude. The value g is then the crisp control action executed.

8.8.3 Medical Sciences

In the field of Medical Science, fuzzy algebra plays an important role. The fuzzy set framework has been utilized in several different approaches to modeling the diagnostic process. In this approach the physician's medical knowledge is represented as a fuzzy relation between symptoms and diseases. Thus, a fuzzy set A of the symptoms observed in the patient and the fuzzy relation R representing the medical knowledge that relates the symptoms in set S to the diseases in set D, then the fuzzy set B of the possible diseases of the patient can be inferred by means of the compositional rule of inference.

i.e. $$B = A \; o \; R$$

or $$\mu_B(d) = \max_{s \in S} [\min (\mu_A(s), \mu_R(s, d))], \quad \forall \; d \in D.$$

This conditional statement is "**If *A* then *B* by *R***".

The membership grades of observed symptoms in fuzzy set A may represent the degree of certainty of the presence of the symptoms or its severity. The membership grade in fuzzy set B denote the degree of certainty with which we can attach each possible diagnostic label to the patient. The fuzzy relations R of medical knowledge should constitute the greatest relation such that given the fuzzy relation Q on the set P of patient and S of symptoms and the fuzzy relation T on the set P of patient and S of symptoms and the fuzzy relation T on the set P of patients and D of diseases, then

$$T = Q \; o \; R$$

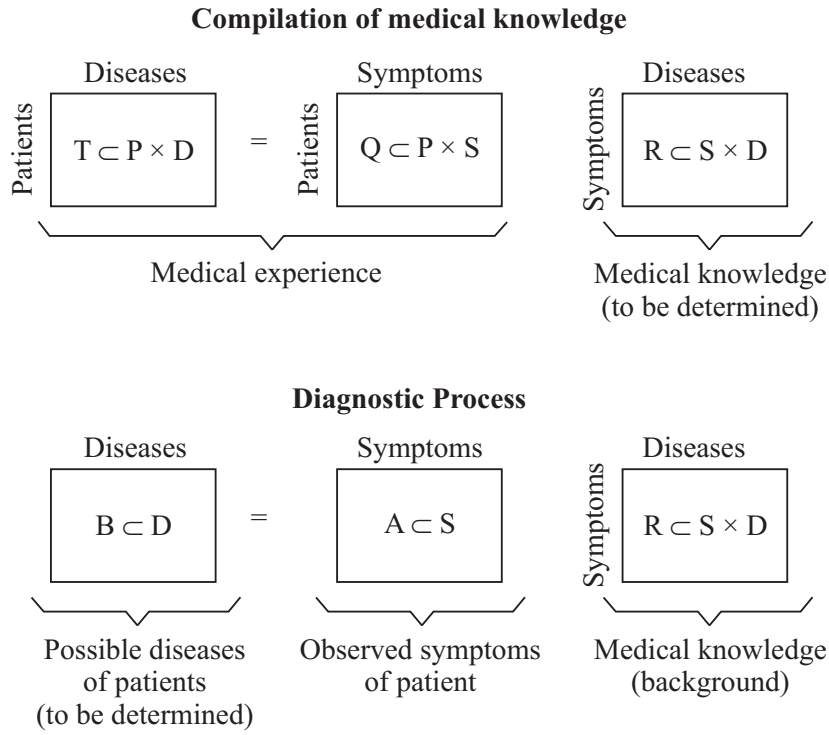


Fig. 8.11: Fuzzy sets and relations in medical diagnosis

8.8.4 Management Sciences and Decision Making Process

Fuzziness can be introduced at various points in the existing model of decision-making and different aspects of management under simple optimization of utility, and uncertainty concerning the states of the system can be handled by modeling the states as well as the utility assigned to each state with fuzzy sets.

A fuzzy model suggested by Bellman and Zadeh in 1970 tells a fuzzy model of decisions that must accomodate certain constraints C and goals G. The membership function then becomes

$$\mu_C : X \to [0, 1]$$

and

$$\mu_G : X \to [0, 1]$$

where X is the universal set of alternative actions.

If the set X of possible action and the set Y of possible effects or outcomes. In this case, the fuzzy constraints may be defined on the set X and fuzzy goals on the set Y such that

$$\mu_C : X \to [0, 1]$$

and $$\mu_G : X \to [0, 1]$$

The function f can be defined as a mapping from the set of action X to the set of outcomes Y,

$$f: X \to Y$$

such that a fuzzy goal G defined on set Y induces a corresponding fuzzy goal G' on set X. Thus

$$\mu_{G'}(x) = \mu_G(f(x))$$

A fuzzy decision D may be defined as the choice that satisfies both the goals G and constraints C. Then

$$D = G \cap C.$$

Then, $$\mu_D(x) = \min [\mu_G(x), \mu_C(x)], \forall\, x \in X.$$

for weighted goals x and constraints m, we get

$$\mu_D(x) = \sum_{i=1}^{n} \mu_i\, \mu_{Gi}(x) + \sum_{j=1}^{m} v_j\, \mu_{Cj}(x)$$

where u_i and v_j are weights attached to each fuzzy goal $G_i (i \in N_n)$ and each fuzzy constraints $C_j (j \in N_m)$, s.t.

$$\sum_{i=1}^{n} u_i + \sum_{j=1}^{m} v_j = 1$$

8.8.5 Computer Science

Fuzzy algebra has extensive feed back in computer science and information/system science. Applications of the mathematics of uncertainty and information within the field of computer science have been particularly in those endeavors concerned with the storage and manipulation of knowledge in a manner compatible with human thinking. This includes the construction of database and information storage and retrieval systems as well as the design of computerized expert systems. The motivation for the application of fuzzy algebra to design a database and information storage and retrieval systems lies in the need of handle information that is less than ideal in the sense of being incomplete, indeterministic, contradictory, vague, imprecise, and so on. The database that can accommodate imprecise information can store and manipulate not only precise facts but also subjective experts opinious, judgements, and values that can be specified in linguistic terms. The fuzzy algebra

also contains other relational operations such as **complement**, **union**, **intersection**, or join, which perform the corresponding tasks on the relation and domains specified in order to produce the desired information.

A computerised expert system models the reasoning process of a human expert within a specific domain of knowledge in order to make the experience, understanding, and problem solving capabilities of the expert available to the nonexpert for purpose of consultation, diagnosis, learning, decision support, or research. An expert system is distingushed from a sophisticated lookup table by the attempt to include the expert system some sense of an understanding of the meaning and relevance of questions and information and an ability to draw non trivial inferences from data. Computerized rule-driven control systems are considered to continue a subset of expert systems, which emulate the reasoning of an expert human operator.

8.9 Uniqueness of Uncertainty Measures

8.9.1 Shannon's Entropy

Uniqueness theorem of Shannon's Entropy

Statement: *If a function* $H : P \to [0, \infty]$ *satisfy the requirements of continuity, weak additinity, monotonicity, branching and normalization, then*

$$H(p_1, p_2, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i \quad \forall\, n \in N.$$

Proof: (*i*) First, we prove the proposition $f(n^k) = kf(n)$ for all $n, k \in N$ by induction, where

$$f(n) = H\left(\frac{1}{n}, \frac{1}{n}, \cdots, \frac{1}{n}\right)$$

is the same function that is used in the definition of weak additivity. For $k = 1$, the proposition is trivially true. By the axiom of weak additivity, we have

$$f(n^{k+1}) = f(n^k \,.\, n) = f(n^k) + f(n).$$

Assume the proposition is true for some $k \in N$. Then,

$$\begin{aligned} f(n^{k+1}) &= f(n^k) + f(n) = kf(n) + f(n) \\ &= (k+1) f(n), \end{aligned}$$

which demonstrates that the proposition is true for all $k \in N$.

(*ii*) Next we can prove that $f(n) = \log_2 n$. We replace function I with our function f. Therefore, we do not repeat the proof here.

(*iii*) We prove now that $H(p, 1-p) = -p \log_2 p - (1-p) \log_2 (1-p)$ for rational p. Let $p = r/s$, where $r, s \in \mathbb{N}$. Then

$$f(s) = H\left(\underbrace{\frac{1}{s}, \frac{1}{s}, \cdots, \frac{1}{s}}_{r}, \underbrace{\frac{1}{s}, \frac{1}{s}, \cdots, \frac{1}{s}}_{s-r}\right)$$

$$= H\left(\frac{r}{s}, \frac{s-r}{s}\right) + \frac{r}{s} f(r) + \frac{s-r}{s} f(s-r)$$

by the branching axiom. By (*ii*) and the definition of p, we obtain

$$\log_2 s = H(p, 1-p) + p \log_2 r + (1-p) \log_2 (s-r).$$

Solving this equation for $H(p, 1-p)$ results in

$$\begin{aligned} H(p, 1-p) &= -p \log_2 r + \log_2 s - (1-p) \log_2 (s-r) \\ &= -p \log_2 r + p \log_2 s - p \log_2 s + \log_2 s \\ &\quad -(1-p) \log_2 (s-r) \\ &= -p \log_2 \frac{r}{s} - (1-p) \log_2 \frac{s-r}{s} \\ &= -p \log_2 p - (1-p) \log_2 (1-p). \end{aligned}$$

(*iv*) We extend now (*iii*) to all real numbers $p \in [0, 1]$ with the help of the continuity axiom. Let p be any real number in the interval [0, 1] and let p' be a series of rational numbers that approach p. Then,

$$H(p, 1-p) = \lim_{p' \to p} H(p', 1-p')$$

by the continuity axiom. Moreover,

$$\lim_{p' \to p} H(p', 1-p') = \lim_{p' \to p} [-p' \log_2 p' - (1-p') \log_2 (1-p')]$$

$$= -p \log_2 p - (1-p) \log_2 (1-p)$$

since all functions on the right-hand side of the equation are continuous.

(*v*) We conclude the proof now by showing that

$$H(p_1, p_2, \ldots, p_n) = -\sum_{i=1}^{n} p_i \log_2 p_i.$$

This is accomplished by induction on n. For $n = 1, 2$, respectively. For $n \geq 3$, we may use the branching axiom to obtain

$$H(p_1, p_2, \ldots, p_n) = H(p_A, p_n) + p_A H\left(\frac{p_1}{p_A}, \frac{p_2}{p_A}, \cdots, \frac{p_{n-1}}{p_A}\right) + p_n H\left(\frac{p_n}{p_n}\right),$$

where $p_A = p_1 + p_2 + \ldots + p_{n-1}$. Since $H(p_n/p_n) = H(1) = 0$ by (*ii*), we have

$$H(p_1, p_2, \ldots, p_n) = H(p_A, p_n) + p_A H\left(\frac{p_1}{p_A}, \frac{p_2}{p_A}, \cdots, \frac{p_{n-1}}{p_A}\right).$$

By (*iv*) and assuming the proposition to be proved is true for $n - 1$, we may rewrite this equation as

$$\begin{aligned} H(p_1, p_2, \ldots, p_n) &= -p_A \log_2 p_A - p_n \log_2 p_n - p_A \sum_{i=1}^{n-1} \frac{p_i}{p_A} \log_2 \frac{p_i}{p_A} \\ &= p_A \log_2 p_A - p_n \log_2 p_n - \sum_{i=1}^{n-1} p_i \log_2 \frac{p_i}{p_A} \\ &= p_A \log_2 p_A - p_n \log_2 p_n - \sum_{i=1}^{n-1} p_i \log_2 p_i + \sum_{i=1}^{n-1} p_i \log_2 p_A \\ &= p_A \log_2 p_A - p_n \log_2 p_n - \sum_{i=1}^{n-1} p_i \log_2 p_i + p_A \log_2 p_A \\ &= -\sum_{i=1}^{n} p_i \log_2 p_i. \end{aligned}$$

■

8.9.2 U-uncertainty

The uniqueness proof is subdivided into four parts, each covered is one section of a derivation of the braveling axiom in terms of basic assignments and its generalization; a determination function U for crisp possibility distributions; a determination of function U for two value possibility distributions; and the uniqueness proof of the U-uncertainty for general possibility distributions. We have to follow this theorem:

The branching axiom expresses the following form in term of basic distributions:

$$U_m(\mu_1, \mu_2, \ldots, \mu_n) = U_m(\mu_1, \mu_2, \ldots, \mu_{k-3}, \mu_{k-2}, + \mu_{k-1}, 0, \mu_k, \mu_{k+1}, \ldots, \mu_n) + (\mu_{k-2} + \mu_{k-1})$$

$$U_m\left(\underbrace{0, 0, ..., 0,}_{k-3} \frac{\mu_{k-2}}{\mu_{k-2} + \mu_{k-1}}, \frac{\mu_{k-1}}{\mu_{k-2} + \mu_{k-1}}, \underbrace{0, 0, \ldots, 0}_{n-k+1}\right)$$

$$-(\mu_{k-2} + \mu_{k-1}) \underbrace{0, 0, \ldots, 0,}_{k-3} \underbrace{0, 0, \ldots, 0,}_{n-k+2} \quad \ldots(1)$$

where $k \in \mathbb{N}_{3, n+1}$.

Proof: Since Eq. (1), formed in terms of basic distributions, which is based on possibility distribution, it is required that $t(p_1, p_2, \ldots, p_n) = (\mu_1, \mu_2, \ldots, \mu_n)$. Let $({}^1\mu_1, {}^1\mu_2, \ldots, {}^1\mu_n)$ denote the unknown basic distribution in the first term on the right-hand side of Eq. (1). This distribution must correspond to the possibility distribution in the first term on the right-hand side, we obtain:

$${}^1\mu_i = \rho_i - \rho_{i+1} = \mu_i \quad \text{for all } i \in \mathbb{N} \text{ except } i = k-2, k-1,$$

$${}^1\mu_{k-1} = \rho_k - \rho_k = 0,$$

and

$${}^1\mu_{k-2} = \rho_{k-2} - \rho_k = (\rho_{k-2} - \rho_{k-1}) + (\rho_{k-1} - \rho_k) = \mu_{k-2} + \mu_{k-1}.$$

Let $({}^2\mu_1, {}^2\mu_2, \ldots, {}^2\mu_n)$ denote the unknown basic distribution in the second term on the right-hand side of Eq. (1). Expressing its connection with the possibility distribution in the corresponding term, we obtain:

$${}^2\mu_i = \rho_i - \rho_{i+1} = 0 \text{ for all } i \in \mathbb{N}_n \text{ except } i = k-2, k-1,$$

$${}^2\mu_{k-2} = 1 - \frac{\rho_{k-1} - \rho_k}{\rho_{k-2} - \rho_k} = \frac{\rho_{k-2} - \rho_{k-1}}{\rho_{k-2} - \rho_k} = \frac{\mu_{k-2}}{\mu_{k-2} + \mu_{k-1}},$$

and

$${}^2\mu_{k-1} = \frac{\rho_{k-1} - \rho_k}{\rho_{k-2} - \rho_k} = \frac{\mu_{k-1}}{\mu_{k-2} + \mu_{k-1}}$$

Similarly, for the basic distribution $3\mu_1, {}^3\mu_2, \ldots, {}^3\mu_n)$ in the last term of Eq. (1), we obtain:

$${}^3\mu_i = \rho_i - \rho_{i+1} = 0 \text{ for all } i \in N_n \text{ except } i = k-2,$$

and $\quad {}^3\mu_{k-2} = 1 - 0 = 1.$

The weight by which the uncertainties in the last two terms of Eq. (1) are multiplied must be derived from the corresponding weight $\rho_{k-2} - \rho_k$. As already shown, $\rho_{k-2} - \rho_k$ corresponds to $\mu_{k-2} - \mu_{k-1}$. This concludes the proof.

Assume now that in formalizing the branching axiom the distinction between ρ_k and ρ_{k+1} in $\boldsymbol{r}$ is ignored by *replacing the lower possibility value* ρ_{k+1} *with the higher value* $\rho_k (k \in N_{n-1})$. Then we obtain an alternative form of the axiom:

$$U(\rho_1, \rho_2, \ldots, \rho_n) = U(\rho_1, \rho_2, \ldots, \rho_{k-1}, \rho_k, \rho_{k+1}, \rho_{k+2}, \ldots, \rho_n) + (\rho_k - \rho_{k+2})\, U\left(\underbrace{1, 1, \cdots, 1}_{k}, \frac{\rho^{k+1} - \rho^{k+2}}{\rho_k - \rho_{k+2}}, \underbrace{0, 0, \cdots, 0}_{n-k-1}\right) - (\rho_k - \rho_{k+2})\, U\left(\underbrace{1, 1, \cdots, 1}_{k+1}, \underbrace{0, 0, \cdots, 0}_{n-k-1}\right) \ldots(2)$$

It can easily be demonstrated, following the line of reasoning in the proof of Theorem 1, that the branching axiom expressed by Eq. (2) has the following form in terms of basic distributions:

$$U_m(\mu_1, \mu_2, \ldots, \mu_n) = U_m(\mu_1, \mu_2, \ldots, \mu_{k-1}, 0, \mu_k + \mu_{k+1}, \mu_{k+2}, \ldots, \mu_n) + (\mu_k + \mu_{k+1})\, U_m\left(\underbrace{0, 0, \cdots, 0}_{k-1}, \frac{\mu_k}{\mu_k + \mu_{k+1}}, \frac{\mu_{k+1}}{\mu_k + \mu_{k+1}}, \underbrace{0, 0, \cdots, 0}_{n-k-1}\right) - (\mu_k + \mu_{k+1})\, U_m\left(\underbrace{0, 0, \cdots, 0}_{k}, \underbrace{0, 0, \cdots, 0}_{n-k-1}\right) \ldots(3)$$

The first generalized form of the branching property involves the grouping of q neighbouring values ($q \geq 2$) of the given basic distribution (rather than two values, as in the branching axioms) by replacing higher values with a single lower value. The form is expressed by the following theorem; the proof of the theorem is done by an introduction on q.

8.9.3 Uniqueness of the U-uncertainty for Two-Value Possibility Distributions

The purpose of this section is to prove the uniqueness of the U-uncertainty for a special class of possibility distributions. These distributions contain only two possibility values in their components, 1 and some value α less than 1 but greater than 0. More formally, a possibility distribution

$\boldsymbol{r} = (\rho_i : i \in \mathbb{N}_n)$ is called a *two-value possibility distribution with u leading* 1s, $u \in \mathbb{N}_{n-1}$, and $n - u$ α's, $0 < \alpha < 1$, if

$$\rho_i = \begin{cases} 1 & \text{for } i \in \mathbb{N}_u \\ \alpha & \text{for } i \in \mathbb{N}_{u+1,\, n,} \end{cases} \quad 0 < \alpha < 1.$$

SUMMARY

1. Membership function is defined by

$$\mu: X \to \{0, 1\}$$

for full membership:

$$\mu(x) = 0$$

Intermediate membership $(0 < \mu(x) < 1)$

and non membership:

$$\mu(x) = 0$$

2. A fuzzy set of a set A is defined by

$$A = \{(x, \mu_A(x): x \in X\}$$

where $\mu_A: X \to [0, 1]$

3. Two fuzzy sets are equal iff

$$\mu_A(x) = \mu_A(x) \; \forall \; x \in X.$$

4. Let X be a set $(\neq \phi)$ and A, B are two fuzzy sets of X with membership function μ_A and μ_B. We can say that the fuzzy set A is **contained** in the fuzzy set B iff

$$\mu_A(x) \leq \mu_B(x) \;\; \forall \; x \in X.$$

5. A fuzzy set of A of set X is called **normal fuzzy set**

iff $\max\limits_{x \to X} \mu_A(x) = 1$

i.e. $\mu_A(x) = 1$ for the least one $x \in X$.

6. The **support** of a fuzzy set A of the set X is the classical set $\{x \in X: \mu_A(x) > 0\}$ denoted by support (A).

7. The **α-cut** or **α-level set** of a fuzzy set A of the set X is crisp set given by

$$A_\alpha = \{x \in X: \mu_A(x) \leq \alpha\}.$$

8. The **complement of a fuzzy set** $A \subseteq X$, we write A^c is defined as

$$\mu_A^c(x) = 1 - \mu_A(x) \; \forall \; x \in X.$$

9. The **union of two fuzzy sets** $A, B \subseteq X$, we write $A \cup B$, is defined as $\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x), \forall\, x \in X$.
   where $\vee$ is a maximum operation.
10. The intersection of two fuzzy sets $A, B \subseteq X$, written as $A \cap B$ is defined as $\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x), \forall\, x \in X$.
    where $\wedge$ is the minimum operator.

EXERCISE

1. Propose and extension of the standard fuzzy set operations (min, max., $1 - a$) to interval valued fuzzy sets.
2. Order the fuzzy sets defined by the following membership grade function ($x \geq 0$) by the exclusion (subset) relation:

   $(i)\ \ \mu_A(x) = \dfrac{1}{1 + 20x}$ $\qquad$ $(ii)\ \ \mu_B(x) = \left(\dfrac{1}{1 + 10x}\right)^{1/2}$

   $(iii)\ \ \mu_C(x) = \left(\dfrac{1}{1 + 10x}\right)^{2}$
3. Explain the term compatibility relation with a suitable example.
4. Define partial ordering in fuzzy relations by taking a suitable example.
5. What do you understand by belief and plausibility measures?
6. What are probability measures in fuzzy algebra.
7. What are the measures of fuzziness? Explain by taking suitable axioms.
8. Explain Shannon Entropy in fuzzy algebraic structures.
9. Explain the term measures of Dissonance.
10. Explain air traffic control system by using fuzzy logic.

Suggested Readings

1. N.Abramson, *Information Theory and Coding*, Mc Graw-Hill, NY (1963)
2. W.R. Ashby, *An Introduction to Cybernetics*, John Wiley, NY [1965]

3. W.R. Garner, *Uncertainty and structure as Psychological concepts*, John Wiley, N.Y. [1986]

4. M.M. Gupta and E.Sanchez eds. *Fuzzy Information and Decision Processes*, North-Holland, NY [1982 *a*]

5. D.S. Jones, *Elementary Information Theory*, Clarendon Press, Oxford [1980]

6. G.J. Klir, *Archetecture of systems Problem solving*, Plenum Press, NY [1985]

7. G.J. Klir, T.A. Folger, *Fuzzy sets, Uncertainty and Information* PHI, N.J. (USA) (1988)

8. D.K.D. Majumder, *Fuzzy Mathematical approach to pattern recognition*, John Wiley, N.Y. [1986]

9. S.K. Yadav, *Discrete Mathematical structures*, Satya Prakashan, N.Delhi [2001]

10. L.A. Zadeh, *Fuzzy sets and Information granularity*, In. Gupta, Ragade, and Yager [1979] *pp*(3 – 18).

❑❑❑

Check for updates

9

Formal Languages and Automata Theory

Stephen Cole Kleene (1909-1994)

Stephen Cole Kleene (1909-1994) was born in Hartford, Connecticut is an academic family. His father was a professor of Economics and mother a poetess. He graduated from Amherst College in 1930 and received his Ph.D, in mathematics from Princeton University in 1934.

A taught at the university of Wisonsin, Madison for 6 years and later on he joined at Amherst College for an year. He served in U.S. Naval Reserve for four years. In 1946, he joined Madison campus and in 1964 became the cyrus C. Mac Duffee professor of mathematics and computer science. He served as chairman of the Department of Mathematics, Acting Director of the mathematics research centre, and Dean of the college of letters and science.

Kleene was awarded by honorary D.SC. by Amherst College in 1970, the steele Prize by Americal Mathematical Society in 1983, National Medal of Science in 1990.

Kleene contributed significantly to the theory of recursive functions, finite Automate theory and finite state machines.

9.1 Introduction

In 1943 W.S. McCulloch and W.Pits investigated and studied finite-state machines and neural networks. In day today life the paradigms of finite-state constructs can be seen everywhere: textiles, traffic signal controlling, automated teller machines, automated telephone service, garage door openers, household appliances, and coin operated machines such as vending machines and slot machines.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_9

Computational models means a formal, **abstract definition of a computer**. By using a model we can analyse more easily, the intrinsic execution time or memory space of an algorithm while ignoring many implementation issues. There are number of models of computation which differ in computational power and the cost of various operations. The following are some computational models:

(*a*) Serial models

(*i*) **Turing Machine:** A model of computation consists of a finite-state machine controller, a read-write head and an unbounded sequential tape. While depending on the current state and symbol read on the tape, the machine can change its state and none the head left or right unless specified, a turing machine is deterministic. Turing machine stores characters on an infinitely long tape, with one square area being scanned by a read/write head at any given time.

(*ii*) **Random Access Machine:** It is a model of computation of which memory consists of an unbounded sequence of registers, each of which may hold an integer. This model allows arithmatic operations to compute the address of a memory register.

(*iii*) **Primitive Recursive Model:** It is based on a total function which can be written, using only nested conditional statements and fixed iteration loops. These models use functions and functions composition to operate on various numbers.

(*iv*) **Cellular Automation:** This model is based on two-dimensional organisation of simple, finite state machines with next state depends on their own state and the states of their eight closed neighbours.

(*v*) **Finite state Machines:** This model consists of a set of states, a start state, an input alphabet and a transition function which maps input symbols and current state to a next state.

The main variants of FSM are:

- Mealy machine
- Moore machine
- Nondeterministic finite state machine
- Recogniser.

(*vi*) **Cell Probe Model:** In this computational model the cost of a computation is measured by the total number of memory accesses to a random access memory, with cell size of *n* bits.

(*vii*) **Pointer Machine:** In this model the memory consists of an unbounded collection of registers or records, connected by pointers. Each of the register contains an arbitrary amount of additional information. To compute, no arithmetic is needed in this address of register.

(*viii*) **Alternating Turing Machine:** It is a nondeterministic turing machine having individual states, from which the machine accepts possible moves of only that state which leads to acceptance.

(*ix*) **Nondeterministic Turing Machine:** It is a turing machine having more than one next state for some combinations of contents of the current cell and current state. Here an input is accepted, if any more sequence leads to acceptance.

(*x*) **Oracle Turing Machine:** It is a turing machine with an extra oracle tape and three extra stages.

(*xi*) **Probabilistic Turing Machine:** A turing machine in which some transitions are of random choices among finitely many alternatives is called probabilistic turing machine. It can be a typical deterministic turing machine.

(*xii*) **Universal Turing Machine:** This turing machine is capable of simulating any other turing machine by encoding the letters.

(*xiii*) **Quantum Computing Machine:** In this type of machine the computation is based on quantum mechanical effects, such as superposition or entanglement, in addition to classical digital manipulations.

(*b*) Parallel Models

(*i*) **Multiprocessor Model:** This model is based on parallel computation which is also based on a set of communicating sequential processors.

(*ii*) **Work-depth Model:** It is a model of parallel computation where one keeps track of the total work and depth of computation, without worrying about how it maps into a turing machine.

(*iii*) **Parallel Random-access Machine:** It is treated as shared memory model of computation where the processors typically execute the same instruction synchronously and access to any memory location occurs in a unit time.

(*iv*) **Shared Memory:** When all the processors have the same global image of the whole memory is termed as shared memory.

(*c*) Abstract Machine: A processor design which is not intended to be implemented as hardware but is the normal executor of a particular intermediate language, used in a procedure for executing a set of instruction in some formal language, possibly by taking an input data and producing the output are called abstract machine. Such abstract machines are not intended to be constructed as hardware and are used in thought experiments about computability. Finite state Machines and Turing machines are the basic examples of abstract machines.

In this chapter we return to the study of formal language and develop another mathematical structure, phase structure grammars, a simple device for construction of useful formal languages. We also examine several popular methods for representing these grammars. As we know, a finite state machine is a model that includes devices ranging from simple "flip-flop" to entire computers. It can also be used to describe the effect of certain computer programs and non-hardware system. Finite state machines are useful in the study of formal languages and are often found in compilers and interpreters for various programming languages. There are several types and extensions of finite-state machines, but in this chapter were are going to introduce only the most elementary versions.

Finite-state machines significantly assist the study of formal languages; for example, a machine can be designed (or a program developed) that determines if an input string over that alphabet $\{a, b\}$ contains *abba* as a substring. (The string *babaabbaba* does, while *abaaba* does not.) This type of machine produces no output, but instead tells whether or not the input string has a certain property.

9.2 Formal Languages

A language consists of a finite or an infinite set of sentences. *Finite Language* can be specified by exhaustively enumerating all their sentences. For infinite languages, such an enumeration is not possible. On the other hand, any device which specifies a language should be

finite. One method of specification which satisfied this requirement uses a generative device is called a grammar. A grammar consists of a finite set of rules or productions which specify the syntax of the language. The study of grammars constitutes an important sub area of computer science which is called *formal language.*

The following are some steps and definitions of formal language:

9.2.1 Equality of Words

Two words $x = x_1\ x_2\ x_3\ \ldots\ x_m$ and $y = y_1\ y_2\ y_3\ \ldots\ y_n$ over Σ^* are equal, denoted by $x = y$, if $m = n$ and $x_i = y_i$ for every i.

The two words are equal if they contain the same number of symbols and the corresponding symbols are the same.

Note: (*i*) Σ^* denotes the set of words over Σ.

(*ii*) A language over Σ is a subset of Σ^* which can be finite or infinite.

(*iii*) A word of length zero is called empty word denoted by λ it contains no symbol.

For example: Let $\Sigma = \{x, y, z, +, -, *, /, \uparrow, (,)\}$, where $*$ denotes multiplication and $\uparrow$ denotes exponentiation. We define a language L over Σ recursively as:

(*i*) $x, y, z \in L$

(*ii*) If u and v are in L, then so are $(+u)$, $(-u)$, $(u + v)$, $(u - v)$, $(u * v)$, (u/v) and $(u \uparrow v)$.

Then L consists of all fully and legally parenthesized algebraic expressions in x, y and z.

For instance $((((x * (x \uparrow z)) - (y * z)) + x) \uparrow z)$ is a fully parenthesized and well-formed algebraic expression.

Note: (*i*) Σ^* contains nonsensical expressions such as $) + x(/y* \uparrow$ etc.

(*ii*) Since both ϕ and $\{\lambda\}$ are subsets of Σ^*, both are languages

(*iii*) The language ϕ is the **empty language**.

(*iv*) The language $\{\lambda\}$ is denoted by $\wedge$.

(*v*) We can emphasize that $\phi \neq \wedge$, since $|\ \phi\ | = 0$ whereas $|\ \wedge\ | = 1$. However, if $\Sigma = \phi$, $\Sigma^* = \wedge$.

Suppose an alphabet Σ contains at least one element a. Then $L = \{a, aa, aaa, \ldots\}$ is an infinite language over Σ. Since $L \subseteq \Sigma^*$, Σ^* is

also infinite. Thus, if $\Sigma \neq \varnothing$, Σ^* contains infinitely many words, each of finite length.

Let z be the concatenation of the words x and y; that is, $z = xy$. Then x is a **Prefix** of z any y a **suffix** of z. For instance, consider the word z = *readability* over the English alphabet; x = read is a prefix of z and y = ability is a suffix of z. Since $x = \lambda x = x\lambda$, every word is a prefix and a suffix of itself; further, λ is both a prefix and a suffix of every word.

The operations of union and intersection can be applied to languages also; after all, languages are also sets. To this end, we extend the definition of concatenation of strings to languages.

9.2.2 Concatenation of Languages

Let A and B be any two languages over Σ. The concatenation of A and B, denoted by AB, is the set of all words ab with $a \in A$ and $b \in B$. *i.e.* $AB = \{ab: a \in A \wedge b \in B\}$.

Let us illustrate this by the example below.

Let $\Sigma = \{0, 1\}$, $A = \{0, 01\}$ and $B = \{\lambda, 1, 110\}$. Then we can find the concatenation AB and BA as under.

- AB consists of strings of the form ab with $a \in A$ and $b \in B$. So

$$\begin{aligned} AB &= \{0\lambda, 01, 0110, 01\lambda, 011, 01110\} \\ &= \{0, 01, 0110, 01, 011, 01110\} \\ &= \{0, 01, 011, 0110, 01110\} \end{aligned}$$

- $$\begin{aligned} BA &= \{ba : b \in B \wedge a \in A\} \\ &= \{\lambda 0, \lambda 01, 10, 101, 1100, 11001\} \\ &= \{0, 01, 10, 101, 1100, 11001\} \end{aligned}$$

The above can be explained by the Fig. 9.1.

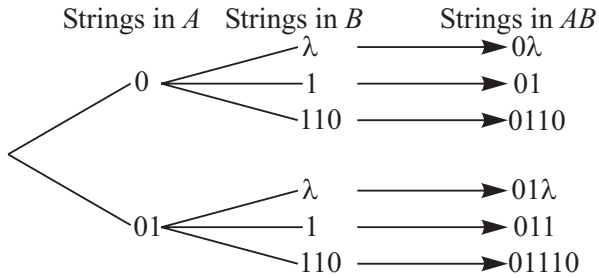


Fig. 9.1

Note: 1. $AB \neq BA$.

2. Further, $|AB| = 5 \leq 6 = 2.3 = |A| . |B|$; whereas $|BA| = 6 = 3.2 = |B| . |A|$. This is so since the word 01 in AB can be obtained in two ways. Therefore, all we can say in general is, if A and B are finite languages, then $|AB| \leq |A| . |B|$.
3. If the languages A and B are the same, then AB is often denoted by A^2. Thus A^2 consists of words obtained by concatenating each word in A with every word in A: $A^2 = \{xy : x, y \in A\}$. More generally, let $n \in N$. Then A^n consists of all words obtained by $n - 1$ concatenations of words in A. In particular, Σ^n denotes the set of words obtained by $n - 1$ concatenations of symbols in Σ, that is, words of length n.

9.2.3 Kleene Closure

Let A be a language over an alphabet Σ. Then $A^* = \bigcup_{n=0}^{\infty} A^n$ is called **Kleene closure** of A. A^* consists of strings obtained by an arbitrary number of concatenations of words from A^* is the **Kleene operator**.

For any language A, $A \subseteq A^*$ *i.e.* when we apply the Kleene operator* on A, the resulting language A^* contains A. We apply * to A^*, we find that $(A^*)\ A^* = A^*$; so we do not get a new language. This explains that A^* is called the Kleene closure of A.

For example: if $A = \{0\}$, $B = \{11\}$, $C = \{000\}$ and $\Sigma = \{0, 1\}$. Then their Kleene closures are

- Since $A = \{0\}$, $A^n = \{0^n\}$. So $A^* = \bigcup_{n=0}^{\infty} A^n = \{0^n : n \geq 0\}$. In other words, A^* consists of strings of zero or more 0's.
- Since $B = \{11\} = \{1^2\}$, $B^2 = BB = \{1111\} = \{1^4\}$. So $B^3 = BB^2 = \{1^2\}\{1^4\} = \{1^6\}$. Thus, in general, $B^n = \{1^{2n}\}$. Thus $B^* = \bigcup_{n=0}^{\infty} B^n = \{1^{2n} : n \geq 0\}$. It consists of words of 1's of even length.
- Since $C = \{0^3\}$, $C^n = \{0^{3n}\}$. So $C^* = \{0^{3n} : n \geq 0\}$, the set of strings of 0's whose lengths are divisible by 3.
- The Kleene closure is the set of all possible words over Σ, namely, Σ^*. (This explains why we denoted it by Σ^* from the beginning of the section.)

SOLVED EXAMPLES

Example 1

In the programming language QUICKBASIC, a numeric variables name must begin with a letter followed by either a period of an alphanumeric character. Explain it.

Solution

Let A = {a, b, …, z} and B = {a, …, z, 0, …, 9, .}. Then concatenation AB gives all numeric variable names containing exactly two characters, namely:

$$aa, ab, \ldots, a0, \ldots, a9, a.$$
$$ba, bb, \ldots, b0, \ldots, b9, b.$$
$$\vdots$$
$$za, zb, \ldots, z0, \ldots, z9, z.$$ ■

Example 2

Let A be a language over Σ. Identify the languages $A\varnothing$ and $A\Lambda$.

Solution

- $A\varnothing = \{ab : a \in A \wedge b \in \varnothing\}$. Since $\varnothing$ contains no elements, no concatenations ab can be performed; therefore, $A\varnothing = \varnothing$. (Similarly, $A\varnothing = \varnothing$.)
- $A\Lambda = A\{\lambda\} = \{a\lambda : a \in A\} = \{a : a \in A\} = A$. (Similarly, $\Lambda A = A$.)

■

Example 3

Let A, B, C and D be any language over an alphabet Σ. Then prove that

(i) $A(B \cup C) = AB \cup AC$. *(ii)* $A(B \cap C) \subseteq AB \cap AC$.

Solution

(*i*) *To prove that* $A(B \cup C) = AB \cup AC$:

We need to show that (*a*) $A(B \cup C) \subseteq AB \cup AC$ and (*b*) $AB \cup AC \subseteq A(B \cup C)$.

- *To prove that* $A(B \cup C) \subseteq AB \cup AC$:

  Let $x \in A(B \cup C)$. Then x is of the form yz, where $y \in A$ and $z \in B \cup C$. If $z \in B$, then $yz \in AB$ and hence $yz \in AB \cup AC$. If $z \in C$, then $yz \in AC$ and therefore $yz \in AB \cup AC$. Thus, in both cases $x = yz \in AB \cup AC$. Consequently, $A(B \cup C) \subseteq AB \cup AC$.

  Therefore, by parts (*a*) and (*b*), $A(B \cup C) = AB \cup AC$.

- *To prove that* $AB \cup AC \subseteq A(B \cup C)$:

  Let $x \in AB \cup AC$. Suppose $x \in AB$. Then $x = ab$ for some $a \in A$ and $b \in B$. Since $b \in B$, b also belongs to $B \cup C$. So $x = ab \in A(B \cup C)$. Similarly, if $x \in AC$, then also $x \in A(B \cup C)$. Thus in both cases $x \in A(B \cup C)$. Consequently, $AB \cup AC \subseteq A(B \cup C)$.

(*ii*) *To prove that* $A(B \cap C) \subseteq AB \cup AC$:

Let $x \in A(B \cap C)$. Then $x = yz$ for some element $y \in A$ and $z \in B \cap C$. Since $z \in B \cap C$, $z \in B$ and $z \in C$. So yz belongs to both AB and AC, and hence $yz \in AB \cap AC$; in other words, $x \in AB \cap AC$. Thus $A(B \cap C) \subseteq AB \cap AC$. ■

Example 4

Let $\Sigma = \{a, b, c\}$, $A = \{a, ab\}$, $B = \{b, ab\}$, *and* $C = \{\lambda, bc\}$.
Verify that (i) $A(B \cup C) = AB \cup AC$. *(ii)* $A(B \cap C) \subseteq AB \cap AC$.

Solution

$$AB = \{ab, aab, abb, abab\}$$
$$AC = \{a\lambda, abc, ab\lambda, abbc\} = \{a, ab, abc, abbc\}$$
$$AB \cup AC = \{a, ab, aab, abb, abc, abab, abbc\}$$
$$AB \cap AC = \{ab, abb\}$$

(*i*) $B \cup C = \{\lambda, b, ab, bc\}$

Then $A(B \cup C) = \{a\lambda, ab, aab, abc, ab\lambda, abb, abab, abbc\}$
$= \{a, ab, aab, abb, abc, abab, abbc\}$
$= AB \cup AC.$

(*ii*) Since $B \cap C = \varnothing$, $A(B \cap C) = \varnothing$ and hence $A(B \cap C) \subseteq AB \cap AC$.

■

Example 5

Let $\Sigma = \{a, b, c\}$, $A = \{a, ab, bc\}$, *and* $B = \{a, bc\}$. *Find* Σ, A^2, *and* B^3.

Solution

- $\Sigma^2 = \{xy : x, y \in \Sigma\} = \{aa, ab, ac, ba, bb, bc, ca, cb, cc\}$
- $A^2 = \{xy : x, y \in A\} = \{aa, aab, abc, aba, abab, abbc, bca, bcab, bcbc\}$
- $B^2 = \{aa, abc, bca, bcbc\}$

So $B^3 = \{aaa, aabc, abca, abcbc, bcaa, bcabc, bcbca, bcbcbc\}$ ■

Example 6

Let A and B be any languages over an alphabet Σ. *Then prove that* $A^*A^* = A^*$.

- *To prove that* $A^* \subseteq A^*A^*$: $\Lambda \subseteq A^*$, $A^*\Lambda \subseteq A^*A^*$

  But $A^*\Lambda = A^*$

  $$A^* \subseteq A^*A^*$$

- *To prove that* $A^*A^* \subseteq A^*$: *Let* $x \in A^*A^*$. *Then* $x = yz$ *with* $y, z \in A^*$. *Since* $y, z \in A^*$., $y \in A^m$ *and* $z \in A^n$ *where* $m, n \in W$. *So* $yz \in A^mA^n = A^{m+n}$. *But* $A^{m+n} \in A^*$, *so* $x = yz \in A^*$. *Thus*

  $$A^*A^* \subseteq A^*$$

Thus, $A^*A^* = A^*$. *

Identify each language over $\Sigma = \{a, b\}$

(i) $\{a, b\}^* \{b\}$ *(ii)* $\{a\}\{a, b\}^*$

(iii) $\{a\}\{a, b\}^*\{b\}$ *(iv)* $\{a, b\}^*\{b\}^*$

Solution

(*i*) $\{a, b\}^*$ consists of all possible words over Σ including λ, whereas $\{b\}$ contains just one word, namely, *b*. Therefore, the language $\{a, b\}^* \{b\}$ consists of words over Σ that have *b* as a suffix.

(*ii*) Similarly, $\{a\}\{a, b\}^*$ consists of words that have *a* as a prefix.

(*iii*) $\{a\}\{a, b\}^* \{b\}$ consists of words that begin with *a* and end in *b*.

(*iv*) Every element in $\{b\}^*$ consists of a finite number of *b*'s. Therefore, $\{a, b\}^* \{b\}^*$ consists of strings followed by a finite number of *b*'s.

Notice that this is different from $\{a, b\}^* \{b\}$. ■

9.3 Grammars

The **grammar** of a language is a set of rules that determines whether or not a sentence is considered valid. For instance, *The milk drinks girl quickly*, although meaningless, is a perfectly legal sentence.

The sentences in a language may be nonsensical, but must obey the grammar. Our discussion deals with only the **syntax** of sentences (the way words are combined), and not with the **semantics** of sentences (meaning). Although listing the rules that govern a natural language such as English is extremely complex, specifying the rules for subsets of English is certainly possible.

The sentence *The girl drinks milk quickly*, has two parts: a subject, The girl, and a predicate *drinks milk quickly*. The subject consists of the definite article *The* and the noun girl. The predicate, on the other hand, consists of the verb *drinks* and the object phrase *milk quickly*; the object phrase in turn has the object *milk* and the adverb *quickly*. This structure of the sentence can appear as a sequence of trees (Figures 9.2-9.6) with the **derivation tree** of the sentence in Figure 9.6.

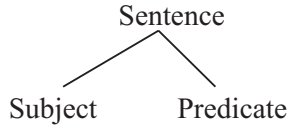


Fig. 9.2

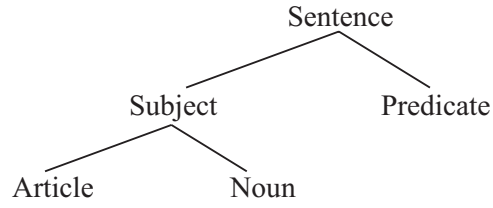


Fig. 9.3

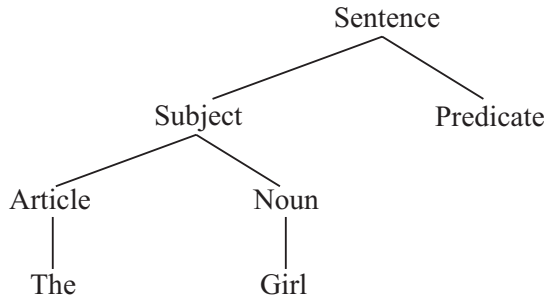


Fig. 9.4

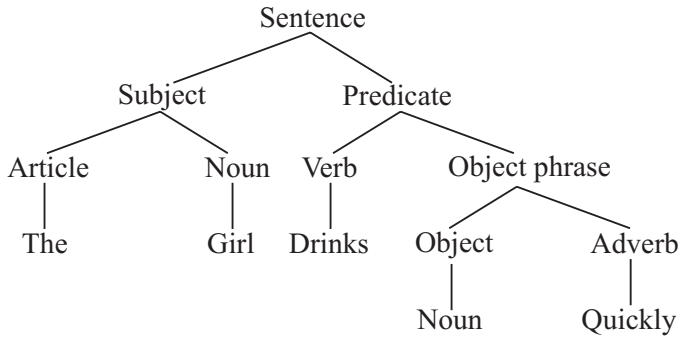


Fig. 9.5

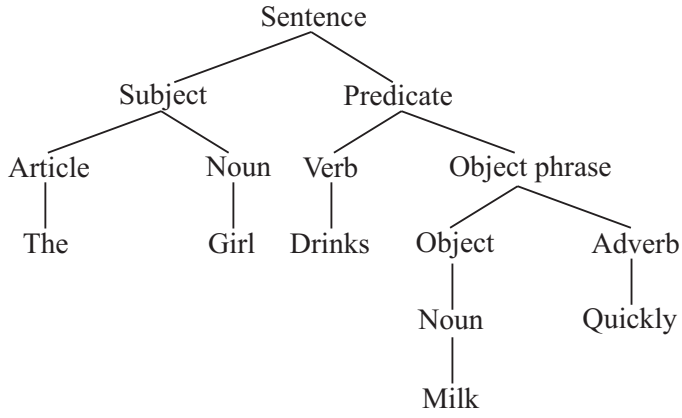


Fig. 9.6

The derivation tree exhibits certain characteristics:

(*i*) Each leaf represents a word, a **terminal symbol**. The set of terminal symbols is $T = \{\text{the, child, drinks, milk, quickly}\}$.

(*ii*) Each internal vertex represents a grammatical class, a **nonterminal**. The set of nonterminals is $N = \{\text{sentence, subject, predicate, article, noun, object phrase, object, verb, adverb}\}$. A nonterminal symbol is enclosed within **angle brackets**, ⟨and⟩. For instance, the nonterminal "subject" is denoted by ⟨subject⟩.

(*iii*) The root of the tree represents the nonterminal symbol ⟨sentence⟩ called the **start symbol**, denoted by σ.

Certain rules can generate the above sentence. Every rule, called a **production rule** or a **substitution rule**, is of the form $w \to w'$ where $w \in N$ and w' may be a terminal symbol, a nonterminal symbol, or a combination of both.

The production rules of the above sentence are:

$$\langle \text{sentence} \rangle \rightarrow \langle \text{sentence} \rangle \ \langle \text{predicate} \rangle$$
$$\langle \text{subject} \rangle \rightarrow \langle \text{article} \rangle \ \langle \text{noun} \rangle$$
$$\langle \text{article} \rangle \rightarrow \textit{the}$$
$$\langle \text{noun} \rangle \rightarrow \textit{girl}$$
$$\langle \text{noun} \rangle \rightarrow \textit{milk}$$
$$\langle \text{predicate} \rangle \rightarrow \langle \text{verb} \rangle \ \langle \text{object phrase} \rangle$$
$$\langle \text{verb} \rangle \rightarrow \textit{drinks}$$
$$\langle \text{object phrase} \rangle \rightarrow \langle \text{object} \rangle \ \langle \text{adverb} \rangle$$
$$\langle \text{object} \rangle \rightarrow \langle \text{noun} \rangle$$
$$\langle \text{adverb} \rangle \rightarrow \textit{quickly}$$

The production rules specify the arrangement of words in a sentence: the **syntax** of the language. They produce syntactically correct sentences (which can be meaningless). For instance, the sentence, *The milk drinks girl quickly* makes no sense but is syntactically valid. Figure 9.7 shows the derivation tree of this sentence.

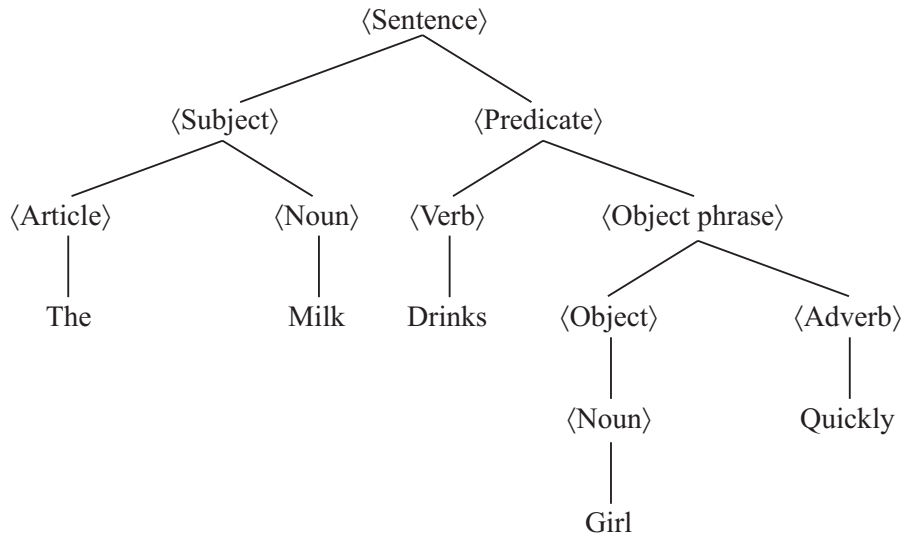


Fig. 9.7

Determining whether a program is syntactically correct is of the utmost importance in computer science. Before executing a program, the complier checks the syntax of each sentence (or expression) by constructing

derivation trees. (This process is **parsing**, and the corresponding derivation tree is a **parse tree**.)

9.3.1 Phase-structure Grammar

A phase-structured grammar G bears four main features *viz*:

(i) A finite set N of nonterminal symbols.

(ii) A finite set T of terminal symbols, where $N \cap T = \phi$.

(iii) A finite subset P of $[(\text{NUT})^* - T^*] \times (N \cup T)^*$
(each element of P is called a production)

(iv) A start symbol σ belonging to N.

$\Rightarrow G = (N, T, P, \sigma)$

Note: These features meet certain requirements:

(i) The start symbol σ is nonterminal.

(ii) No symbol can be both terminal and nonterminal.

(iii) Every production has at least one nonterminal symbol on its LHS, because $P \subseteq [(N \cup T)^* - T^*] \times (N \cup T)^*$. Also, P is a binary relation from $(N \cup T)^* - \text{T}^*$ to $(N \cup T)^*$

(iv) If $(w, w') \in P$, we then write $w \rightarrow w'$; since $w \in (N \cup T)^* - T^*$, w contains at least one nonterminal symbol; but $w' \in (N \cup T)^*$; so it may contain terminal symbols, nonterminals or both.

9.3.2 Derivations of Grammar

Let $G = (N, T, P, \sigma)$ be a grammar.

If $\text{w} = x\alpha y$ and $w' = x\beta y$ are any two words in $(N \cup T)^*$, and there exists a production $\alpha \rightarrow \beta$ then the word w' is called **directly derivable** from w; we write then, $w \Rightarrow \text{w}'$.

If there is a finite sequence of words $w_0, w_1, w_2, \ldots, w_n$ in $(N \cup T)^*$ s.t. $w_0 \Rightarrow w_1, w_1 \Rightarrow w_2, \ldots, w_{n-1} \Rightarrow w_n$.

Then w_n is derivable from w_0.

The finite sequence of steps $w_0 \Rightarrow w_1 \Rightarrow \ldots \Rightarrow w_n$ is called derivation of w_n from w_0.

The set of words in T^* derivable from σ and G is the **language generated** by G (denoted by $L(G)$).

For example we can identify the language $L(G)$ generated by grammar for $N = \{A, B, \sigma\}$, $T = \{a, b\}$ and $P = \{\sigma \to aA, A \to bA, A \to a\}$.

Then $G = (N, T, P, \sigma)$ is a grammar. The production $A \to bA$ is recursive.

Since the grammar contains exactly one production involving σ, namely, $\sigma \to aA$, start with it to find every word in the language. Now select the next production: $A \to bA$ or $A \to a$. The production $A \to a$ produces exactly one word, a. $A \to bA$ chosen n times, produces $\sigma \Rightarrow aA \Rightarrow abA \Rightarrow ab^2A \Rightarrow \ldots \Rightarrow ab^nA$. Now $A \to a$ yields the word ab^na and, when $n = 0$, this yields $a\lambda a = aa$. (*Note*: $b_0 = \lambda$, the null word.) Every word derivable from σ fits the form ab^na, where $n \geq 0$. In other words, $L(G) = \{ab^na \mid n \geq 0\}$, we can redefine a grammar $G = (N, T, P, \sigma)$ that generate the language $L = \{a^n\, b^n : n \geq 1\}$

Since every word in L must contain the same number of a's and b's. G must contain a production of the form $\sigma \to aAb$. Consequently, to produce a new word from aAb containing the same number of a's and b's requires another production $A \to aAb$. From these two productions, we can derive all strings of the form $a^n\, Ab^n$. All that remains to be done to define the grammar is the production $A \to \lambda$ to terminate the recursive procedure. Thus, $N = \{\sigma, A\}$, $T = \{a, b, \lambda\}$, and $P = \{\sigma \to aAb, A \to aAb, A \to \lambda\}$.

The first two production rules in this example look quite similar, except for the start symbol, and can be combined into a single production, $\sigma \to a\sigma b$. The production rules $\sigma \to aAb$ and $A \to \lambda$ can yield the word ab, so the third production is $\sigma \to ab$. Thus $P' = \{\sigma \to a\sigma b, \sigma \to ab\}$ is an additional production set that yields the same language. In other words, the grammars $G = \{N, T, P, \sigma\}$ and $G' = \{N', T', P', \sigma\}$ generate the same language L, where $N' = \{\sigma\}$ and $T' = \{a, b\}$. Thus $L(G) = L(G')$, so the grammars G and G' are **equivalent**. Our conclusion: *The grammar that generates a language need not be unique.*

9.3.3 Backus-Normal Form (BNF) or Backus Naur Form

In Backus-Naur Form (in the name of J.W. Backus and Peter Naur) The production symbol $\to$ is denoted by :: = ; The production combined by separating their RHS with vertical bars. For instance, the productions $w \to w_1, w \to w_2, \ldots, w \to w_n$ become $w ::= w_1 / w_2 / \ldots / w_n$.

John W. Backus (1924-) was born in Philadelphia. He received his B.S. and M.S. in mathematics from Columbia University. After joining IBM in 1950, he became instrumental in the development of FORTRAN and ALGOL (ALGOrithmia Language). He received the W. W. McDowell Award from The Institute of Electrical and Electronics Engineers (IEEE) in 1967, the National Medal of Science in 1975, the A. M. Turing Award from the Association for Computing Machinery in 1977, the Harold Pender Award from the University of Pennsylvania in 1983, and an honorary doctorate from York University, England, in 1985.

Peter Naur (1928-), a computer scientist and prolific writer, was born in Frederiksberg, Denmark. After receiving his M.A. in astronomy from Copenhagen University in 1949, he spent the next two years at Cambridge University, England, where he used the EDSAC, one of the earliest computers, to pursue astronomy. He received his Ph.D. in astronomy from Copenhagen in 1957.

From 1953 to 1959, he consulted for the design of the first Danish computer, the DASK. Beginning around 1964, he became increasingly involved in datalogy (a word he coined), the study of data and data processes. In 1963, Naur was given the Hagemanns Gold Medal and three years later the Rosenhjaer Prize.

For example: We study the following production rules:

$$\begin{aligned}
\langle \text{sentence} \rangle &\rightarrow \langle \text{subject} \rangle \ \langle \text{predicate} \rangle \\
\langle \text{subject} \rangle &\rightarrow \langle \text{article} \rangle \ \langle \text{noun} \rangle \\
\langle \text{predicate} \rangle &\rightarrow \langle \text{verb} \rangle \ \langle \text{object} \rangle \\
\langle \text{object} \rangle &\rightarrow \langle \text{article} \rangle \ \langle \text{noun} \rangle \\
\langle \text{article} \rangle &\rightarrow \textit{a} \\
\langle \text{article} \rangle &\rightarrow \textit{the} \\
\langle \text{noun} \rangle &\rightarrow \textit{hare} \\
\langle \text{noun} \rangle &\rightarrow \textit{tortoise} \\
\langle \text{noun} \rangle &\rightarrow \textit{race} \\
\langle \text{verb} \rangle &\rightarrow \textit{beats} \\
\langle \text{verb} \rangle &\rightarrow \textit{wins}
\end{aligned}$$

BNF shortens these rules:

$$\begin{aligned}
\langle \text{sentence} \rangle &::= \langle \text{subject} \rangle \ \langle \text{predicate} \rangle \\
\langle \text{subject} \rangle &::= \langle \text{article} \rangle \ \langle \text{noun} \rangle \\
\langle \text{predicate} \rangle &::= \langle \text{verb} \rangle \ \langle \text{object} \rangle \\
\langle \text{object} \rangle &::= \langle \text{article} \rangle \ \langle \text{noun} \rangle \\
\langle \text{article} \rangle &::= \textit{a} \mid \textit{the} \\
\langle \text{noun} \rangle &::= \textit{hare} \mid \textit{tortoise} \mid \textit{race} \\
\langle \text{verb} \rangle &::= \textit{beats} \mid \textit{wins}
\end{aligned}$$

Note: The grammar for the language of correctly nested parentheses contains one production:

⟨nested parentheses⟩ :: = λ | (⟨nested parentheses⟩)

where λ denotes the null string. [Using this definition, you may verify that (()) and ((())) are valid nested parentheses, whereas (() and ())) are not.]

Illustration: An integer is a string of digits preceded by an optional sign, + or –. Using BNF, it can be defined as follows:

⟨integer⟩ :: = ⟨signed integer⟩ | ⟨unsigned integer⟩

⟨signed integer⟩ :: = ⟨sign⟩ | ⟨unsigned integer⟩

⟨sign⟩ :: = + | –

⟨unsigned integer⟩ :: = ⟨digit⟩ | ⟨digit⟩ ⟨unsigned integer⟩

⟨digit⟩ :: = 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

For instance, 234, + 234, and –234 are valid integers. Figure 9.8 shows the derivation tree for the integer +234.

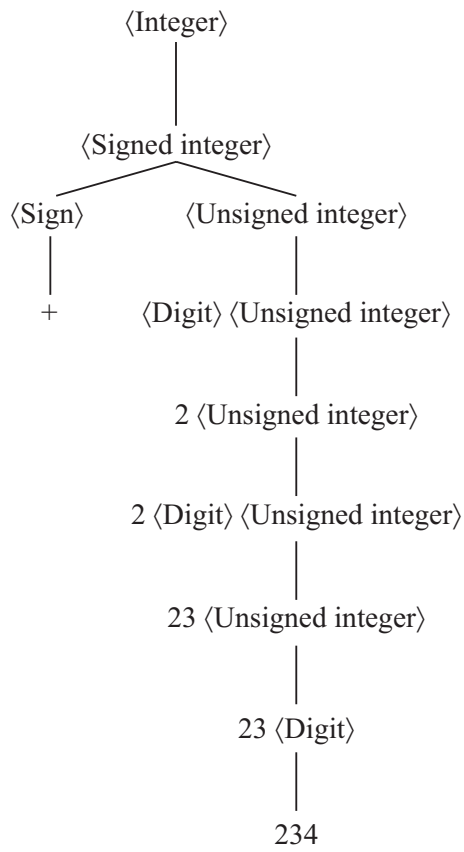


Fig. 9.8

Note: The grammar defined in this example is $G = (N, T, P, \sigma)$, where:

(*i*) $N = \{\langle\text{integers}\rangle, \langle\text{signed integer}\rangle, \langle\text{unsigned integer}\rangle, \langle\text{sign}\rangle, \langle\text{digit}\rangle\}$,

(*ii*) $T = \{+, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$,

(*iii*) The production rules are:

$$\langle\text{integer}\rangle \rightarrow \langle\text{signed integer}\rangle \mid \langle\text{unsigned integer}\rangle$$
$$\langle\text{signed integer}\rangle \rightarrow \langle\text{sign}\rangle \mid \langle\text{unsigned integer}\rangle$$
$$\langle\text{sign}\rangle \rightarrow + \mid -$$
$$\langle\text{unsigned integer}\rangle \rightarrow \langle\text{digit}\rangle \mid \langle\text{digit}\rangle\ \langle\text{unsigned integer}\rangle$$
$$\langle\text{digit}\rangle \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$

(*iv*) The start symbol σ is integer.

9.3.4 Chomsky Grammar

Named ofter Noam Chomsky who developed **context-sensitive, Context-Free and Regular Grammars.**

Let $G = (N, T, P, \sigma)$ be a grammar. Let $A, B \in N$ and $\alpha, \alpha' \beta \in (N \cup T)^*$.

Notice that α, α', and β could be the null word.

- Any phrase-structure grammar is **type 0**.
- G is **context-sensitive** (or **type 1**) if every production is of the form $\alpha A\alpha' \rightarrow \alpha\beta\alpha'$.
- G is **context-free** (or **type 2**) if every production is of the form $A \rightarrow \alpha$.
- G is **regular** (or **type 3**) if every production is of the form $A \rightarrow t$ or $A \rightarrow tB$, where $t \in T$.

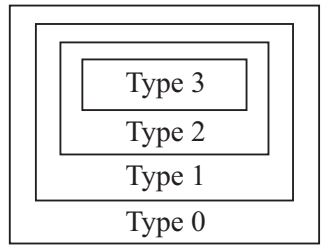


Fig. 9.9: Chomsky hierarchy of grammars

(Avram) Noam Chomsky (1928-), a linguist, writer, and political activist, was born in Philadelphia, as the son of a Hebrew scholar. At 10 he proofread the manuscript of his father's edition of a 13th century Hebrew grammar. "This backdoor introduction to 'historical linguistics' had considerable impact on his future" (The New York Times Magazine). The young Chomsky, however, was more passionate about politics than about grammar. On graduating from Central High School in Philadelphia in 1945, Chomsky entered the University of Pennsylvania and received his B.A. in 1949 and M.A. 2 years later. Chomsky received his Ph.D. in linguistics from the University of Pennsylvania in 1955 and joined the faculty at the Massachusetts Institute of Technology. (MIT)

In 1966, Chomsky became the Ferrari P. Ward Professor of Modern Languages and Linguistics. He had been a visiting professor at Columbia, Princeton, and the University of California at Los Angeles and at Berkeley.

For example: Consider the grammar $G = (N, T, P, \sigma)$, where $N = \{A, B, \sigma\}$, $T = \{a, b\}$ and $P = \{\sigma \rightarrow a\sigma b, \sigma \rightarrow aAb, aAb \rightarrow aBb, A \rightarrow a, B \rightarrow b, A \rightarrow \lambda, B \rightarrow \lambda\}$. In the production $aAb \rightarrow aBb$, A can be replaced with B only if A is surrounded by a and b. Notice that $L(G) = \{a^m b^m, a^m b^{m+1}, a^{m+1} b^m \mid m \geq 1\}$.

9.3.5 Ambiguous Grammar

A grammar G is ambiguous if a string in $L(G)$ has more than one derivation tree.

The following grammar G defines the syntax of simple algebraic expressions:

$$\langle \text{expression} \rangle ::= \langle \text{expression} \rangle \ \langle \text{sign} \rangle \ \langle \text{expression} \rangle \mid \langle \text{letter} \rangle$$
$$\langle \text{sign} \rangle ::= + \mid -$$
$$\langle \text{letter} \rangle ::= a \mid b \mid c \mid \ldots \mid z$$

This grammar can produce the expression $a - b + c$ two ways, as the derivation trees in Figure 9.10 show. As a result, G is an ambiguous grammar. The following are simplified production rules for an *if-then statements S:*

$$S ::= \text{if } \langle \text{expression} \rangle \text{ then } \langle \text{statement} \rangle \mid$$
$$\text{if } \langle \text{expression} \rangle \text{ then } \langle \text{statement} \rangle \text{ else } \langle \text{statement} \rangle$$
$$\langle \text{expression} \rangle ::= E_1 \mid E_2$$
$$\langle \text{statement} \rangle ::= S_1 \mid S_2 \mid \text{if } \langle \text{expression} \rangle \text{ then } \langle \text{statement} \rangle$$

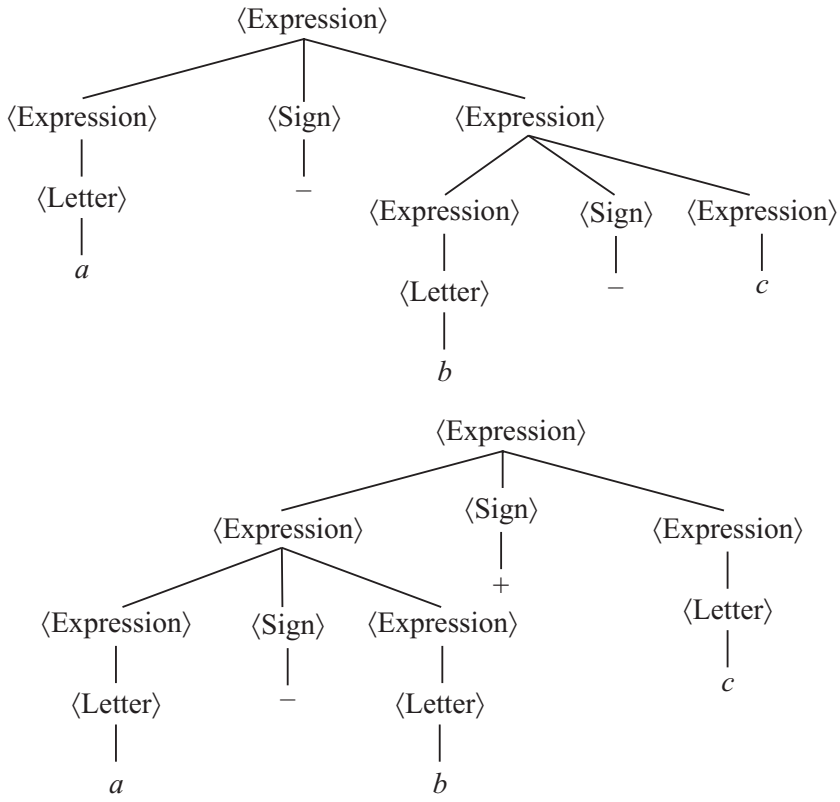


Fig. 9.10

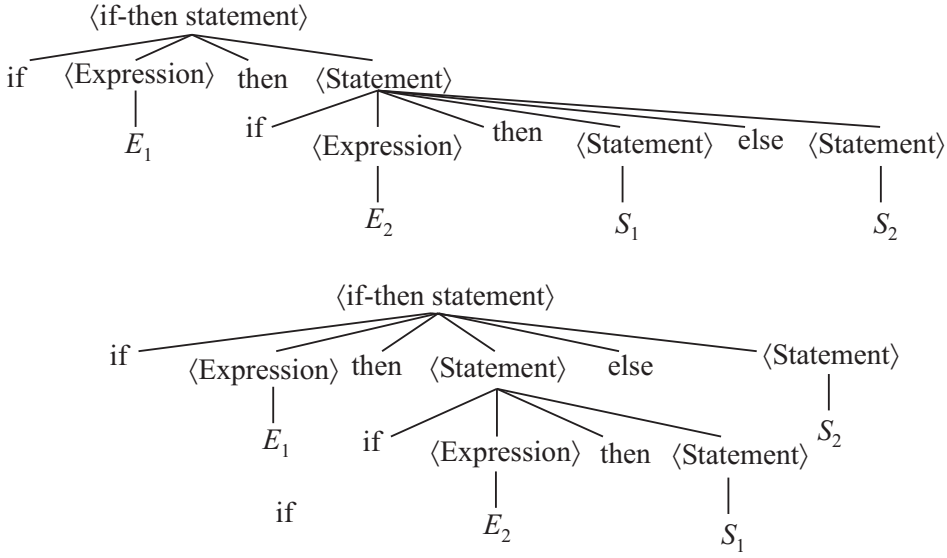


Fig. 9.11

To see that these rules produce an ambiguous grammar, notice that the if-then statement

$$\textit{If } E_1 \textit{ then if } E_2 \textit{ then } S_1 \textit{ else } S_2 \qquad \ldots(i)$$

can be interpreted in two ways:

(*i*) If E_1 then (if E_2 then S_1 else S_2), or

(*ii*) If E_1 then (if E_2 then S_1) else S_2.

Using indentation, these possibilities can be displayed as follows:

```
(i) if E1 then
        if E2 then
            S1
        else
            S2
```

```
(ii) if E1 then
         if E2 then
             S1
     else
         S2
```

two options:

```
if E1 then
    if E2 then
        S1
    else
else
     S2
```

```
if E1 then
          begin
            if E2 then
                S1
          end
else
    S2
```

9.4 Finite-State Automation (FSA)

An automation is a system which obtains, transforms, transmits and uses information to perform its functions without direct human participation. It is self operational. In general, it is a machine that has an input tape, that can be put into any of the several states. Various symbols are written on the tape before execution. The automation begins reading the symbols on the tape from left to right. Upon reading a symbol from the tape, the machine changes its state and advances the tape. After reading the inputs, the machine halts.

Definition: A **finite-state automation** (FSA), *M*, manifests five characteristics:

- A finite set, *S*, of **states** of the automation.
- A specially designated state, s_0, called the **initial state**.

- A subset A of S, consisting of the **accepting states** (or **final states**) of the automation.
- A finite set, I, of **input symbols**.
- A function f: $S \times I \rightarrow S$, called the **transition function** or the **next-state function**.

In symbols, $M = (S, A, I, f, s_0)$.

Some Practical Applications

9.4.1 Counting to Five

It is a finite state machine, describing a process of counting upto five. The transition to state 5 is a loop back to itself as shown in Fig. 9.12

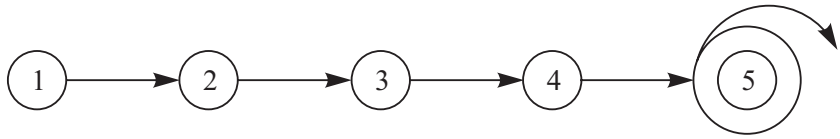


Fig. 9.12: Process of counting

9.4.2 Process of Getting up in the Morning (Alarm)

In this system, more than one arrow emerge from a particular circle *i.e.* either transition can occur. We notice the two edger leaving the '**Alarm Buzzing**' state. This tells that each time when the alarm goes on the morning, the person might choose to the alarm off and gets up, or he might hit the snooze button and go back to sleep. (refer Fig. 9.13).

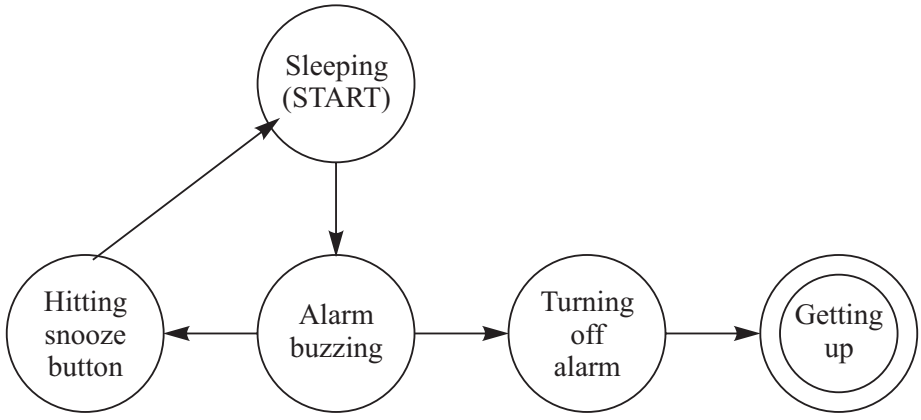


Fig. 9.13

9.4.3 Traffic Light

Consider the FSM for a traffic light as shown in Fig. 9.14. We can deduce that all traffic lights start off by showing red, then make a transition to a green light and finally display a yellow light before returning to the red state.

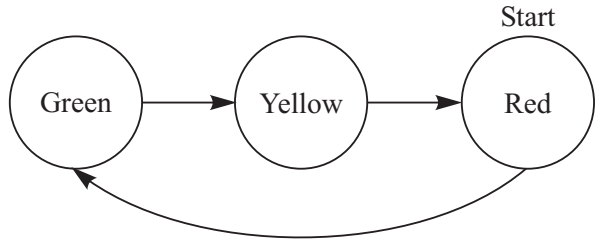


Fig. 9.14: Traffic light model

Traffic Model can be drawn a state machine to characterized a traffic light that might change into a flashing red mode at some point, and then later on changing back into the normal green-yellow-red cycle.

To solve, there are two ways to approach this problem:

(*i*) to add a state that indicates 'flashing red'

(*ii*) to add a state that means dark.

Which then alternates with red to appear as flashing red. (As shown in Fig. 9.15)

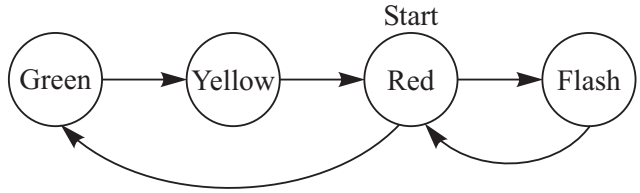


Fig. 9.15: Traffic light modeling with flash state

for 'dark' state, we solve as.

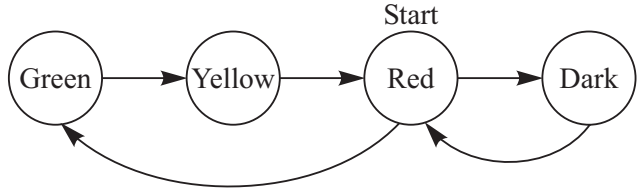


Fig. 9.16: Traffic light modeling with Dark state

9.4.4 Vending Machine

We consider the operation of a soft drink vending machine which charges 15 cents for a car. Preted that we are a machine. Initially we are waiting for a customer to come and put some coins, *i.e.* we are in the **waiting-for-customer state**. We assume that only nickels and dimes are used, when a customer comes and puts in the first coin, say a drive, we are no longer in the waiting for customer state. We have received 10 cents and are waiting for more coins to come. So, we might say that we are in the 10 cents state. If the customer puts is a nickel, then we have received 15 cents and we wait for the customer to select the soft drink. So, we are in another state *i.e.* 15-cents state. When the customer selects soft drinks, we must give the a can of soft drink. After that we stay is that state, until another coin is put in, to start the process afresh or we may terminate the operation and start from the initial state. The states and the transitions between them, for this vending machine, can be represented with the Fig. 9.17.

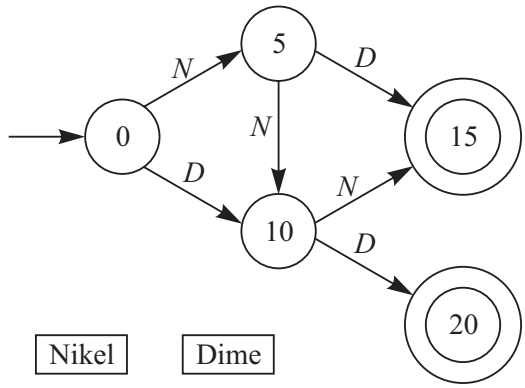


Fig. 9.17: A Vending Machine Model

Here *D*s on arrows represent a dime and *N*s, a Nickel. Assume that the machine terminate its operation when it receives 15 cents or more.

9.5 Finite-State Machine (FSM)

A **finite-state machine** (FSM), M, bears six characteristics:

- A finite set, S, of **states**;
- A finite **input alphabet**, I;
- A finite set, O, of **output symbols**;
- A **transition function**, $f : S \times I \to S$;

- A **output function**, $g : S \times I \rightarrow O$;
- A **initial state**, s_0.

In symbols, $M = (S, I, O, f, g, s_0)$.

For example: Let $S = \{s_0, s_1, s_2\}$, $I = \{a, b\}$, and $O = \{0, 1\}$. Define functions $f : S \times I \rightarrow S$ and $g : S \times I \rightarrow O$ by means of Table 9.1. For example, $f(s_0, b) = s_1$, $f(s_2, b) = s_1$, $g(s_0, b) = 1$, and $g(s_2, b) = 1$.

Table 9.1

	f		*g*	
S ***I***	***a***	***b***	***a***	***b***
s_0	s_0	s_1	0	1
s_1	s_1	s_2	1	0
s_2	s_2	s_1	1	1

Then $M = (S, I, O, f, g, s_0)$ is an FSM with transition function f and output function g. Table is the **transition table** of the machine.

Like an FSA, and FSM can be represented by a **transition diagram**, with one main difference: every directed edge (s_j, s_k) has two labels. One indicates the input symbol i; the other the output o from entering i into state s_j. For instance, if $f(s_j, i) = s_k$ and $g(s_j, i) = 0$, the directed edge (s_j, s_k) is labeled $i/0$.

For example: Construct the transition table of the FSM in Figure 9.18.

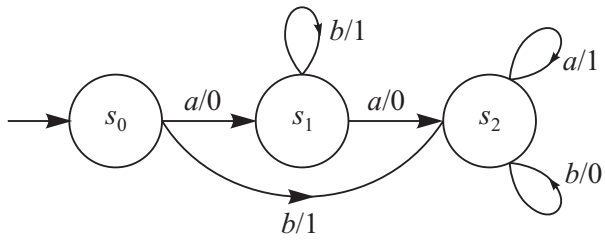


Fig. 9.18

Solution

From the transition diagram, $f(s_0, a) = s_1$, $f(s_0, b) = s_2$, $f(s_1, a) = s_2$, $f(s_1, b) = s_1$, $f(s_2, a) = s_2$, and $f(s_2, b) = s_2$, also $g(s_0, a) = 0$, $g(s_0, b) = 1$, $g(s_1, a) = 0$, $g(s_1, b) = 1$, $g(s_2, a) = 1$, and $g(s_2, b) = 0$. These values generate the transition table in Table 9.2.

Table 9.2

	f		*g*	
S \ *I*	***a***	***b***	***a***	***b***
s_0	s_1	s_2	0	1
s_1	s_2	s_1	0	1
s_2	s_2	s_2	1	0

9.6 Finite-State Automata

(Deterministic and Nondeterministic Finite-State Automata)

9.6.1 Deterministic Finite-State Automata (DFSA)

In an FSA $M = (S, A, I, f, s_0)$, where $| I | = m$, exactly m outgoing edges leave every state s_i, each labeled with a unique element of I. Besides, since $f : S \times I \to S$, every state-input pair yields a unique state; in other words, every state-input pair uniquely determines the next state.

The language of DFSA is regular. We have to prove this as under:

Let $M = (S, A, I, f, s_0)$ be a DFSA and $L(M)$ denote the language accepted by automaton. We shall construct a regular grammar G using the machine M and show that $L(G) = L(M)$.

To construct the grammar $G = (N, T, P, \sigma)$, choose $N = S$ as the set of states, $T = I$ as the input alphabet, and $\sigma = s_0$ as the initial state. Define the productions P this way:

Let s_i and s_j be any two states, and x any input symbol. If $f(s_i, x) = s_j$, an accepting state, include the production $s_i \to x$. Clearly, G is a regular grammar.

To prove that $L(M) \subseteq L(G)$:

Let $x = x_1 x_2 \ldots x_n$ be a string accepted by the automaton M; that is, let $x \in L(M)$. Then the transition diagram of the automaton contains a directed path s_0–s_1–s_2–…–s_n, where s_n is an accepting state. Correspondingly, these production rules follow:

$$s_0 \to x_1 s_1 \qquad \ldots(i)$$
$$s_1 \to x_2 s_2$$
$$\vdots$$
$$s_{i-1} \to x_i s_{o^i}$$
$$\vdots$$
$$s_{n-1} \to x_n \qquad \text{(**Note:** } s_n \text{ is an accepting state.)}$$

and the derivation of the string x:

$$\Rightarrow x_1 s_1 \qquad \ldots(ii)$$
$$\Rightarrow x_1 x_2 s_2$$
$$\vdots$$
$$\Rightarrow x_1 x_2 \ldots x_{n-1} s_{n-1}$$
$$\Rightarrow x_1 x_2 \ldots x_{n-1} x_n$$

since $s_{n-1} \to x_n$. Thus $x \in L(G)$, so $L(M) \subseteq L(G)$.

Conversely, let $x = x_1 x_2 \ldots x_n \in L(G)$.

(*i*) Correspondingly, the transition diagram of the automaton M must contain a directed path, s_0–s_1–s_2–…–s_n. The string determined by this path is $x = x_1 x_2 \ldots x_n$. Since the last production in the derivation.

(*ii*) $s_{n-1} \to x_n$, s_n must be an accepting state, thus $x \in L(M)$ and hence $L(G) \subseteq L(M)$.

Thus $L(M) = L(G)$. In other words, the language accepted by the DFSA is regular.

9.6.2 Nondeterministic Finite-State Automata

A **nondeterministic finite-state automaton** (NDFSA) M exhibits five characteristic.

- A finite set S of **states**;
- A specially designated state σ, called the **initial state**;
- A subset A of S consisting of the **accepting states** (or **final states**) of the automaton;
- A finite set I of **input symbols**;
- A function $f : S \times I \to P(S)$, called the **transition function** (or the **next-state function**). [*Note*: $P(S)$ denotes the power set of S.]

  In symbols, $M = (S, A, I, f, \sigma)$.

For example: The NDFSA $M = (S, A, I, f, \sigma)$, where $S = \{\sigma, A, B, C\}$, $A = \{F\}$, $I = \{a, b\}$, and f is defined by Table 9.3 below. Its transition diagram is given in Figure 9.19.

Table 9.3

S \ I	*a*	*b*
σ	{σ, *A*}	∅
A	{*A*}	{*B*, *F*}
B	{*F*}	{*B*}
F	∅	∅

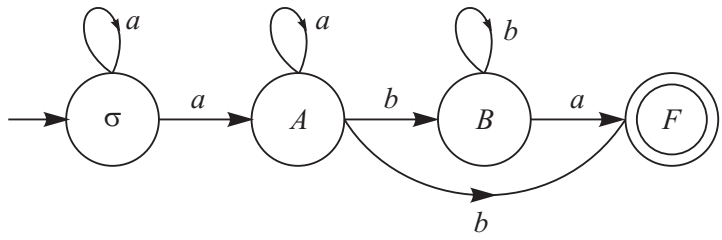


Fig. 9.19

9.6.3 Equivalent Nondeterministic Finite State Automata

A string is **accepted** or **recognized** by a NDFSA $M = (S, A, I, f, s_0)$ if a directed path runs from the initial vertex s_0 to an accepting state that generates the string. The language of all strings accepted by M is $L(M)$. Two NDFSAs are **equivalent** if they accept the same language.

For example: The string a^2b^3 a is accepted by the NDFSA in Figure 9.19. Two paths generate it, σ-σ-*A*-*B*-*B*-*B*-*F* and σ-*A*-*A*-*B*-*B*-*B*-*F*. The automaton accepts strings $a^m\, b$ and $a^m\, b^n a$, where $m, n \geq 1$. Thus $L(M) = \{a^m\, b, a^m\, b^n a \mid m, n \geq 1\}$.

Every regular language is accepted by an NDFSA, to prove this: Let $G = (N, T, P, \sigma)$ be a regular grammar. Make a suitable NDFSA $M = (S, A, I, f, s_0)$ such that $L(G) = L(M)$. Select $I = T$, $s_0 = \{\sigma\}$, and N as the set of nonaccepting states of M. Since the grammar contains productions of the form $s_i \to x$, introduce an accepting state F; choose $S = N \cup \{F\}$ and $A = \{F\}$. Finally, since every production of G is $s_i \to xs_j$ or $s_i \to x$, the transition function $f: S \times I \to P(S)$ follows: $f(s_i, x) = \{s_j \mid s_i \to xs_j\} \cup \{F \mid s_i \to x\}$.

$$L(G) = L(M).$$

SOLVED EXAMPLES

Example 1

Determining if an input string over the alphabet {a, b} contains abba as a substring involves the following five steps:

Solution

Step 0: If the first symbols in the string is *a*, move to step 1 and look for the character *b*. Otherwise, no progress has been made.

Step 1: If the next character is *b*, the substring *ab* has occurred, so go to step 2 and look for another *b*. Otherwise, the symbol *b* is still missing, so stay in step 1.

Step 2: If the next symbol is *b*, the substring *abb* exists; go to step 3; if *a*, return to step 1.

Step 3: If the next symbol is *a*, the given input string contains the substring *abba*; otherwise, return to step 0 and start all over again.

Step 4: Once the substring *abba* has occurred in the input string, any sequence of *a*'s and *b*'s may follow.

These steps can be represented by a digraph (See Figure 9.20), each vertex representing a step. Exactly two edges, labeled *a* or *b*, leave each vertex.

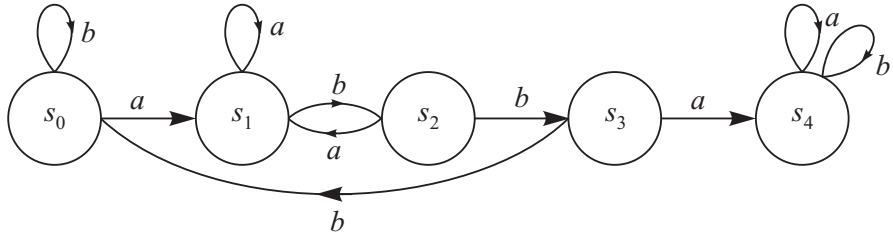


Fig. 9.20

To determine the action required from a given step, simply follow the directed edges from the corresponding vertex. For example, at vertex s_3 (step 3) if the next input symbol is *a*, move to vertex s_4 (step 4); otherwise, return to vertex s_0 (step 0). The other (labeled) edges are interpreted similarly.

The digraph indicates a string contains *abba* as a substring if and only if the directed path the string determines terminates at vertex s_4. The string *abab* determines the path s_0-s_1-s_2-s_1-s_2, which does not end at s_4; consequently, *abab* is not acceptable. On the other hand, the string *ababbab* determines the path s_0-s_1-s_2-s_1-s_2-s_3-s_4-s_4, which terminates at s_4; so the string does have the desired property. ■

Example 2

Draw the transition diagram of the FSA M = (S, A, I, f, s_0), where S = {s_0, s_1, s_2, s_3, s_4}, A = {s_2}, I = {a, b, c}, and f is defined by Table 9.4 on next page 411.

Table 9.4

S \ I	*a*	*b*	*c*
s_0	s_1	s_2	s_3
s_1	s_4	s_2	s_3
s_2	s_1	s_4	s_3
s_3	s_1	s_2	s_4
s_4	s_4	s_4	s_4

Solution

The automaton contains five states, with s_2 the only accepting one. Since there are three input symbols, three edges originate from every state. Draw a directed edge from states s_i to stat s_j if there exists an input symbol x such that $f(s_i, x) = s_j$. For instance, $f(s_1, c) = s_3$, so a directed edge labeled c runs from states s_1 to state s_3. Figure 9.21 displays the resulting transition diagram, where, for convenience, the three loops at s_4 appear as a single loop with labels a, b, and c.

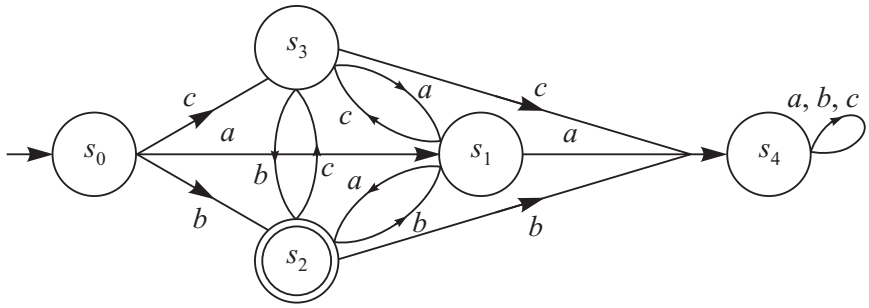


Fig. 9.21 ■

Example 3

After a bank customer inserts his bank card into the ATM, it requests him to input his secret identification number (ID). Suppose the ID is 234. Design an FSA that models the ATM.

Solution

The input to the automaton contains three digits d. It has five states: s_0 (the initial state, waiting for the first digit in the ID), s_1 (the first digit is correct; now waiting for the second digit), s_2 (the second digit is correct; waiting for the third digit), s_3 (the third digit is correct), and s_4 (the trap state that captures all invalid ID's). The ensuring FSA is shown in Figure 9.22.

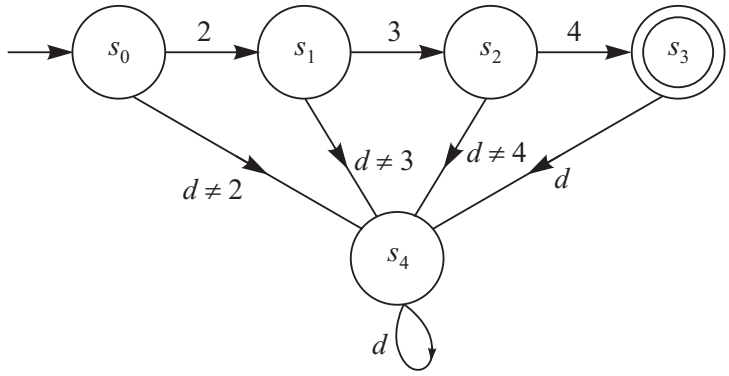


Fig. 9.22 ■

Example 4

Let I = O = {0, 1}. A ***unit delay machine****, an FSM M = (S, I, O, f, g, s_0), delays and input string by unit time. When the string $x_1x_2...x_n$ is input, it produces $0x_1x_2...x_n$ as the output. Construct such a machine.*

Solution

Since each state has two possible outputs, each has two outgoing edges. The machine must certainly have an initial state s_0. With the first output always 0, both edges leaving s_0 must yield 0. The machine must remember whether the previous input was 0 or 1; this requires two additional states, s_1 and s_2. If the previous input was 0, the machine moves to state s_1 and outputs 0; if it was 1, it moves to states s_2 and outputs 1. Figure 9.23 shows the transition diagram of this FSM.

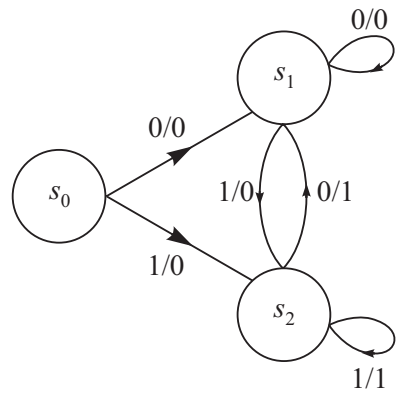


Fig. 9.23

For instance, the output 101110 yields the output 010111 (Verify this.), which has lost the trailing zero of the input. By appending a 0, however, to the input, that is, by inputing 1011100, the desired output result: 0101110. Deleting the leading 0 yields an exact copy of the input.

■

Example 5

Show that the language $L = \{a^n b^n : n \geq 1\}$ is not regular.

Proof (by contradiction)

Suppose L is regular. Then a DFSA M exists such that $L(M) = L$. Suppose M has m states. Since the string $x = a^{m+1}\, b^{m+1} \in L$, x is accepted by the DFSA. Let P be the path corresponding to x; it ends at an accepting state F.

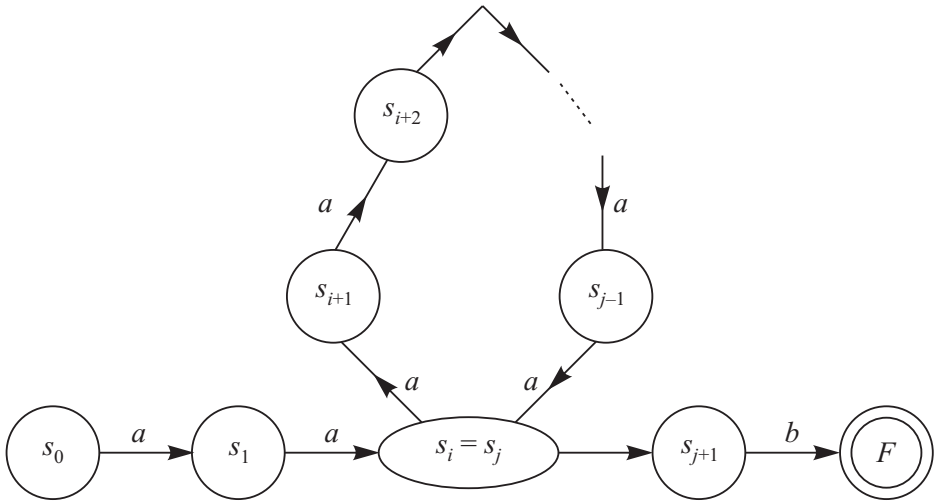


Fig. 9.24

The path corresponding to the substring a^{m+1} contains $m + 1$ states. But, since only m states exist, by the pigeonhole principal at least two of the $m + 1$ states, say, s_i and s_j, where $i < j$, must be the same; consequently, there must be a directed cycle at s_i, each edge labeled a (see Figure 9.24). Let l be the length of the cycle. The path s_0-s_1-…-s_i-s_{j+1}-s_{j+2}-…-F generates the string $x' = a^{m+1-l}\, b^{m+1}$. Since this path ends at F (an accepting state), x' is accepted by the automaton; so $x' \in L$. This is a contradiction, since x' does not contain the same number of a's and b's. Thus L is not a regular language.

SUMMARY

1. A **formal language** over an alphabet Σ is a subset of Σ*, the set of all possible words over Σ.
2. The **concatenation** of two languages A and B over Σ consists of words ab with $a \in A$ and $b \in B$.

3. $A^n = \{a_1 a_2 \ldots a_n \mid a_i \in A\}$, where $A^0 = \Lambda$.
4. $A^* = \bigcup_{n=0}^{\infty} A^n$ is the **Kleene closure** of a language A.
5. A **grammar** $G = (N, T, P, \sigma)$ consists of a finite set N of **nonterminal symbols**, a finite set T of **terminal symbols**, a finite set P of **production rules**, and a **start symbol** σ.
6. A word $w' = x\beta y$ is **directly derivable** from a word $w = x\alpha y$ if a production $\alpha \rightarrow \beta$ exists; we then write $w \Rightarrow w'$. A word w_n is **derivable** from w_1 if there exists a finite sequence of **derivations**, $w_1 \Rightarrow w_2$, $w_2 \Rightarrow w_3, \ldots, w_{n-1} \Rightarrow w_n$. The language derivable from σ is the **language generated** by G, denoted by $L(G)$.
7. In BNF, each production $w \rightarrow w'$ is written as $w ::= w'$; each nonterminal symbol w is enclosed by angle brackets, as in $\langle w \rangle$; and production rules with the same left-hand sides are combined with vertical bars into a single rule.
8. A **type 0** grammar has phrase-structure.
9. In a **context-sensitive** (**type 1**) grammar, every production is of the for $\alpha A \alpha \rightarrow \alpha \beta \alpha'$.
10. In a **context-free** (**type 2**) grammar, every production is of the form $A \rightarrow \alpha$.
11. In a **regular** (**type 3**) grammar, every production is of the form $A \rightarrow t$ or $A \rightarrow tB$.
12. A language $L(G)$ is **context-sensitive**, **context-free**, or **regular** according as whether G is context-sensitive, context-free, or regular.
13. An **ambiguous** language contains a word that has more that one derivation tree.
14. The language accepted by a DFSA is regular.
15. A **FSM** $M = (S, A, I, f, s_0)$ consists of a finite set S of **states**, a finite set A of **accepting states**, a finite set I of **input symbols**, a **transition function** $f: S \times I \rightarrow S$, and an **initial state** s_0. Every state-input pair yields a unique next-state of the automaton.
16. A **transition table** defines the transition function.
17. A **transition diagram** can represent a DFSA. The initial state s_0 is identified by drawing an arrow toward it; and accepting state by two concentric circles around it.

18. An input string is **accepted** by an automaton M if an only if the string traces a path that ends at an accepting state. The **language** $L(M)$ **accepted** by M consists of all words recognized by it.

19. Two automata M and M' are **equivalent** if $L(M) = L(M')$.

20. An **FSM** $M = (S, I, O, f, g, s_0)$ consists of a finite set S of states, a finite set I of **input symbols**, a finite set O of **output symbols**, a **transition function** $f : S \times I \to S$, an **output function** $g : S \times I \to O$, and an **initial state** s_0. Every state-input pair produces a next-state and an output value.

21. A **transition table** can define the transition and output functions of an FSM.

22. A **transition diagram** also can define an FSM.

23. An **NDFSA** $M = (S, A, I, f, s_0)$ consists of a finite set S of states, a subset A of S of accepting states, a finite set I of input symbols, a transition function $f : S \times I \to P(S)$, and an initial state σ. A state-input pair may be paired with zero, one, or more states.

24. Every regular language is accepted by a NDFSA.

25. Every NDFSA is equivalent to a DFSA.

EXERCISE

1. Let $\Sigma = \{-, ₹, a, b, \ldots, z, 0, \ldots, 9,\}$. Define a language L of legal identifies in Jave recursively.

2. Let $A = \{a, b\}$, $B = \{a, b, ab\}$, $C = \{c\}$ and $D = \{c, bc\}$, verify that

(a) $\wedge A = A$

(b) $A(B \cup C) = AB \cup AC$

(c) $(B \cup C)A = BA \cup CA$

(d) If $A \subseteq B$ and $C \subseteq D$, then $AC \subseteq BD$.

3. Prove that (True or false)

(a) $\wedge = \phi$ (b) $A\phi = \phi$

(c) $A\phi = \phi A$ (d) $A\wedge = \Lambda$

(e) $|A \times B| = |B \times A|$

4. Design an FSA accepting words over $\{a, b\}$ that begin with aa and end in bb.

5. Find the grammar of the regular language accepted by the parity check machine.

[**Hint:** N = $\{E, O\}$, $T = \{a, b\}$, $S = \{E\}$, and the production rules are:

$E \to aO$, $E \to bE$, $O \to aE$, $O \to bO$, $E \to b$, and $O \to a$]

The regular grammar defined by the parity check machine M is $G = (N, T, P, S)$. [So $L(G) = L(M)$ = the set of strings over T containing an even number of a's.]

6. Consider the following language over $A = \{a, b\}$'

(*a*) $L_1 = \{a^m b^n : m > 0, n > 0\}$.

(*b*) $L_2 = \{b^m a b^m : m > 0, n > 0$

(*c*) $L_3 = \{a^m b^m : m > 0\}$.

7. Find a regular expression r over $A = \{ab\}$ s.t. $L_1 = L(r)$ $\forall$ $i = 1, 2, 3, \ldots$.

[**Ans.** (*a*) a^* abb^*, (*b*) bb^* abb^*, (*c*) no regular expression]

8. Let S be a finite set, and $L \subseteq S^*$, then prove that L is a regular set iff $L = L(G)$ for some regular grammar $G = (V, S, V_0, \to)$.

9. Write grammars for the following languages:

(*a*) The set of nonnegative odd integers

(*b*) The set of nonnegative even integers with no leading zeros permitted.

10. Give a grammar which generates

$L = \{w \mid w$ consists of an equal number of a's and b's$\}$

11. Give a context-free grammar which generates

$L = \{w : w$ contains twice as many 0s as 1s$\}$.

12. Construct a regular grammar which will generate all strings of 0s and 1s having both an odd number of 0s and an odd number of 1s.

13. Obtain a grammar for the language

$L = \{0^i 1^j : i \neq j$ and $i, j > 0\}$.

14. Obtain a context-sensitive grammar for the language $\{am^2 : m \geq 1\}$.

15. Construct a context-sensitive grammar for the language $\{w : w \in \{a, b, c\}^* - \{L\}$, where w contains the same number of a's, b's and c's$\}$.

16. Determine if Fig. 7.25 is a DFSA.

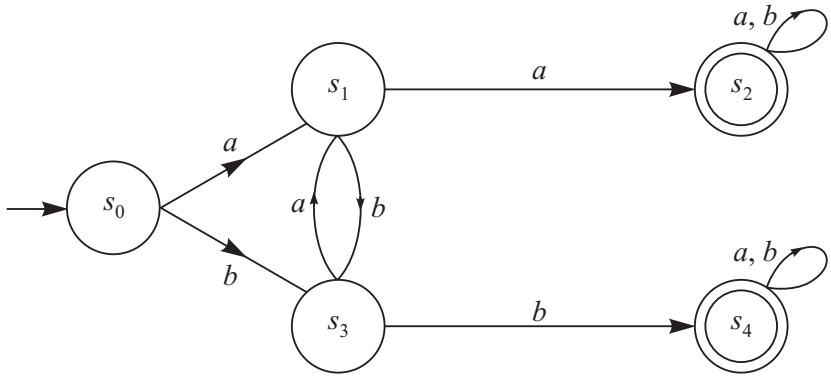


Fig. 7.25

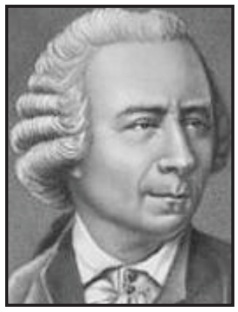
Suggested Readings

1. B. Hayes, "*On the Finite-State Machine, a Minimal Model of Mousetraps,* Ribosomes, and the Human Soul," *Scientific American*, Vol. 249 (Dec. 1983), pp. 19-28, 178.
2. J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, Reading, MA, 1979.
3. Z. Kohavi, *Switching and Finite Automata Theory*, 2nd ed., McGraw-Hill, New York, 1978.
4. P. Linz, *An Introduction to Formal Languages and Automata*, D. C. Heath, Lexington, MA 1990.
5. J. C. Martin, *Introduction to Languages and the Theory of Computation*, 2nd ed., McGraw-Hill, New York, 1997.
6. M. Sipser, *Introduction to the Theory of Computation*, PWS, Boston, 1997.
7. W. A. Wulf *et al., Fundamental Structures of Computer Science*, Addison-Wesley, Reading, MA, 1981, pp. 1-64.
8. P. K. Srimano, Nasir SFB, *A textbook on Automata Theory*, Cambridge University Press, 2008.

❑❑❑

10

The Basics of Graph Theory



Leonhard Euler (1707-1783)

Leonhard Euler (1707-1783) was born in an intellectual family in Basel, Switzerland. His father was a mathematician and a calvinist paster and wanted him to be a pastor in future. Euler had different ideas, he followed his father´s wishes and entered in the university of Basel to study Hebrew and theology. For his hard work and remarkable ability the famous mathematician Johann Bernoulli was very much impressed and recognized this young chap´s talent. He approached Euler´s father to change the track and got a positive response. Euler was allowed to pursue his studies in mathematics.

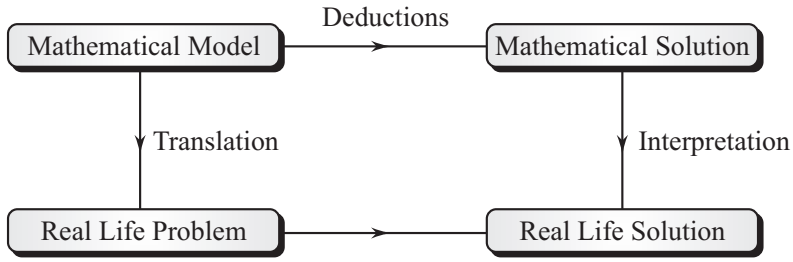
Euler brought out his first paper. He won Paris Academy prize 12 times. He was the most prolific mathematician and contributed every branch of mathematics. With his phenominal memory, he had perfect recall for every result. As a genius, he could work anywhere and under any conditions. Euler belongs in a class by himself. Euler is known as the father of Graph theory.

10.1 Introduction

In the last four decades graph theory has established itself as a worthwhile mathematical discipline and there are many applications of graph theory to a wide variety of subjects that include Operations Research, Physical Sciences, Economics, Genetics, Sociology, Linguistics, Engineering and Computer Science etc. such a development may be roughly described as follows

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_10



Graph theory also has been independently discovered many times through some puzzles that arose from the physical world, consideration of chemical isomers, electrical networks etc. The graph theory has been independently discovered many times, since it may be considered as a counter part of Applied Mathematics.

In the words of Sylvester, "the theory of ramification is one of pure colligation, for it takes no account of magnitude or position, geometrical lines are used, but have no more real bearing on the matter than those employed in genealogical tables have in explaining the laws of procreation."

In the present century, there have already been a great many rediscoveries of graph theory which we can only mention most briefly in this chronological account.

Graph theory is considered to have begun in 1736 with the publication of Euler's solution of the Königsberg bridge problem. Euler (1707–1782) is known as the father of graph theory as well as topology.

There are various types of graphs, each with its own definition. Unfortunately, some people apply the term "graph" rather loosely, so we cannot be sure what type of graph we are talking about unless we ask them. After we have finished this chapter, we expect us to use the terminology carefully not loosely. To motivate the various definitions we will take suitable example.

10.2 Graph! What is it?

10.2.1 Simple Graph

A simple graph G is a pair $G = (V, E)$

Where

- V is a finite set, called the vertices of G, and
- E is a subset of $P_2(V)$ (a set E of two-element subset of V), called the edges of G.

To avoid rotational ambiguities, we always assume tactly that $V \cap E = \phi$.

A nuisance in first learning graph theory is that there are so many definitions. They all correspond to intuitive ideas, but can take a while to absorb. Some ideas have multiple names. For example, graphs are sometimes called networks, vertices are sometimes called nodes, and edges are sometimes called areas. Even worse, no one can agree on the exact meanings of terms. For example, in our definition, every graph must have at least one vertex. However, other authors permit graphs with no vertices. (The graph with no vertices is the single, stupid counterexample to many would–be theorems–so we're banning it!). This is typical; everyone agrees more-or-less what each terms means, but disagrees about weird special cases. So do not be alarmed if definitions here differ subtly from definitions we see elsewhere. Usually, these differences do not matter.

If we take an example of graph as shown in Fig. 10.1

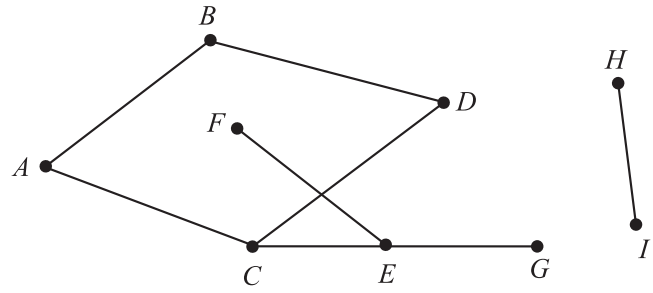


Fig. 10.1: Simple Graph

The vertices correspond to the dots in the picture, and the edges correspond to the lines. Thus, the dots and lines diagram above in a pictorial representation of the graph (V, E) where

$V = \{A, B, C, D, E, F, G, H, I\}$

$E = \{\{A, B\}, \{A, C\}, \{B, D\}, \{C, D\}, \{C, E\}, \{E, F\}, \{E, G\}, \{H, I\}\}$

Hereafter, we use A—B to denote an edge between vertices A and B rather than the set notation $\{A, B\}$. Note that A—B and B—A are the same edge, just as $\{A, B\}$ and $\{B, A\}$ are the same set.

Two vertices in a graph are said to be adjacent if they are joined by an edge, and an edge is said to be incident to the vertices it joins. The number of edges incident to a vertex is called the degree of the vertex.

For example, in the graph above, A is adjacent to B and B is adjacent to D, and the edge A—C is incident to vertices A and C. Vertex H has degree 1, D has degree 2, and E has degree 3.

Deleting some vertices or edges from a graph leaves a subgraph. Formally, a subgraph of $G = (V, E)$ is a graph $G' = (V', E')$ where V' is a nonempty subset of V and E' is a subset of E. Since a subgraph is itself a graph, the end points of every edge in E' must be vertices in V'.

Example 1

A Computer Network Problem

Computers are linked with one another so that they can interchange information through a particular network. Given a collection of computers, we can describe this linkage in clear terms so that we might answer the questions like; "How can we send a message from computer A to computer B using the fewest possible intermediate computers?"

We can do this by making a list coasting of pairs of computer that are interconnected. The pairs are unordered since, if computer C can communicate with computer D, then the reverse in also true. We have implicitly assumed that the computers are distinguished from each other. It is sufficient to say that "*APC* is connected to Ayan." We must specify which *PC* and which Ayan. Thus, each computer has a unique identification label of some sort.

For those who like pictures rather than lists, we can put dots on a piece of paper, one for each computer we label each dot with a computer's identifying label and draw a curve connecting two dots if the corresponding computers are connected. The shape of the curve does not matter because we are only interested in whether two computers are connected or not. We can see two such pictures of the same graph (as shown in Fig. 10.2). Each computer has been labeled by initials of its owner.

Computer (vertices) are indicated by dots (·) with labels. The connections (edges) are indicated by lines. When limes cross, they might be thought of as cables that live on top of each other not as cables that are joined.

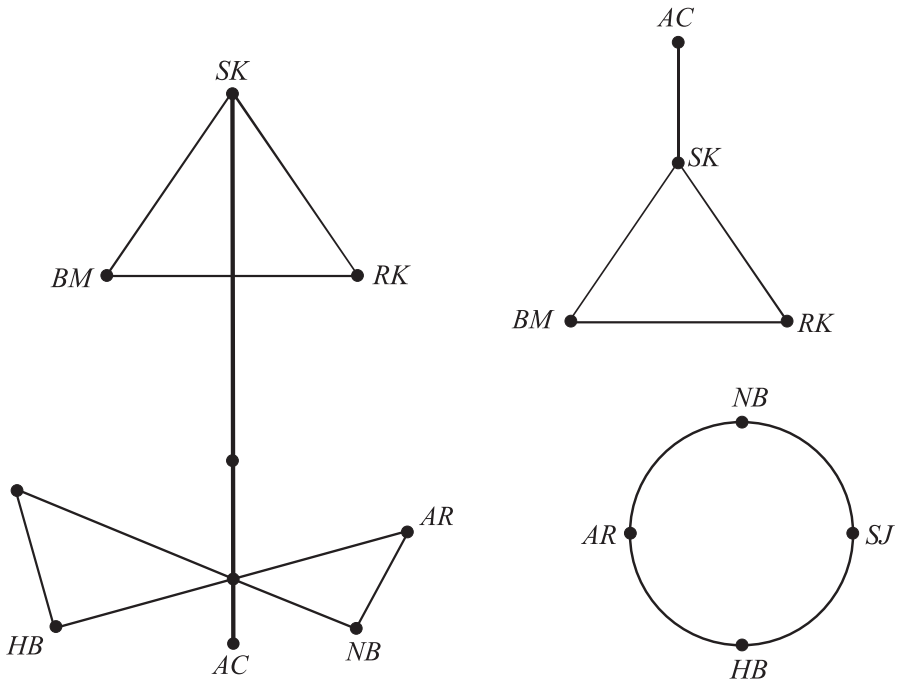


Fig. 10.2: Computer Network Problem ■

10.2.2 Graph

A graph is a triple $G = (V, E, \phi)$

Where

- *V is a finite set, called the vertices of G.*
- *E is a finite set, called the edges of G and*
- ϕ *is a function with domain E and Co-domain* $P_2(V)$.

This definition tells us that to specify a graph G it is necessary to specify the sets V and E and the function ϕ. We have to specify V and ϕ in set theoretic terms.

The function ϕ is sometimes called the incidence function of the graph. The two element of $\phi(x) = \{u, v\}$, for any $x \in E$, are called the vertices of the edge x, and we say u and v are joined by x. We can also say that u and v are adjacent vertices and u adjacent to v or, equivalently, v is adjacent to u.

For any $u \in v$, if v is a vertex of an edge x than we can say the x is incident on v. Likewise, we can say v is the member of x, v is on x, or v is in x. Of course, v is a member of x actually means v is a member of $\phi(x)$.

Example 2

Problem of Routes between Cities (Geographical Placement Problem)

We consider four cities named, with characteristic mathematical charm, A, B, C and D. Between these cities there are various routers of travel, denoted by a, b, c, d, e, f and g (as shown in Fig. 10.3).

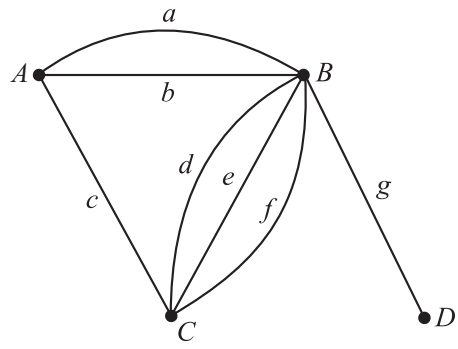


Fig. 10.3: Geographical Placement Problem

On observing this picture, we can see that there are three roots between cities B and C. These routs are d, e and f. This picture is intended to give us only information about the interconnections between the cities. This leaves out so many aspects of the situation that might be an interest to the traveler. Unlike a typical map, no claim is made that the picture represents in any way the distance between the cities or their geographical placement relative to each other. The object shown in this picture is a graph.

One is tempted to list the pairs of the cities which are connected; in other words, to extent a simple graph from the information. This does not describe the problem adequately because there can be more than one route connecting a pair of cities *e.g.* d, e and f connecting cities B and C as shown in Fig. 10.3.

In the pictorial representation of the cities graph

$G = (V, E, \phi)$ where

$V = \{A, B, C, D\}$, $E = \{a, b, c, d, e, f, g\}$.

and $$\phi = \begin{pmatrix} a & b & c & d & e & f & g \\ \{A, B\} & \{A, B\} & \{A, C\} & \{B, C\} & \{B, C\} & \{B, C\} & \{B, D\} \end{pmatrix}$$

The function ϕ is determined from the picture by comparing the name attached to a route with the two cities connected by that route. Thus, the route name d is attached to the route with endpoints B and C. This means that $\phi(d) = \{B, C\}$

Since part of the definition of a function includes its co-domain and domain, ϕ determined $P_2(V)$ and E. Also, V can be determined from $P_2(V)$. Consequently, we could have said a graph is a function ϕ whose domain is a finite set and co-domain is $P_2(V)$ for some finite set V. Instead, we choose to specify V and E explicitly because the vertices and edges play a functional role in thinking about graph G.

Fig 10.4 Shows two additional pictures of the same cities graph given above (Fig. 10.3)

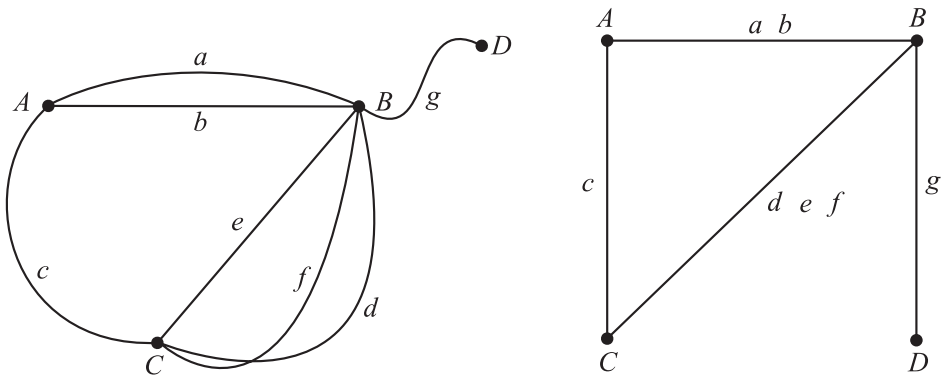


Fig. 10.4: Same Cities Location Graph

The drawings look very different but exactly the same set V and function ϕ are specified in each case. It is very important that are understand exactly what information is needed to completely specify the graph. When thinking in terms of cities and routes between them, we want the pictorial representation of the cities to represent their geographical positioning. If the pictorial representation does this, that's fine, but it is not a part of the information required to define a graph. Geographical location is extra information. The geometrical positioning of the vertices, A, B, C and D is very different, in the first of the two pictorial representations above, than it was in our original representation of the cities. However, in each of these cases, the vertices on a given edge are the same and hence the graphs specified are the same. In the second of two pictures above, a different method of specifying the graph is given. There, ϕ^{-1} the inverse of ϕ, is given. For example, $\phi^{-1}(\{C, B\})$ is shown to be $\{d, e, f\}$. Knowing

ϕ^{-1} determines ϕ and hence determines G since the vertices A, B, C and D are also specified. ■

10.2.3 Loops

A loop is an edge which connects a vertex to itself. Graphs and simple graphs cannot have loops, why? Let us explain it. Suppose $e \in$ E is a loop in a graph that connects $v \in V$ to itself. Then $\phi(e) = \{u, v] = \{v\}$ because repeated elements in the description of a set count only once. They are the same element.

Since $\{v\} \notin P_2(V)$, the range of ϕ, we cannot have $\phi(e) = \{u, v\}$ *i.e.* we cannot have a loop.

If we want to allow loops, we will have to change our definitions. For a graph, we expand the codomain of ϕ to be $P_2(V) \cup P_1(V)$. For a simple graph we need to change the set of allowed edges to include loops. This could be done by saying that E is a subset of $P_2(V) \cup P_1(V)$ instead of a subset of just $P_2(V)$. For example, if $V = \{1, 2, 3\}$ and $E = [\{1, 2\}, \{2\}, \{2, 3\}]$, this simple graph has a loop at vertex 2 and vertex 2 is connected by edge to the other two vertices. When we want to allow loops, we can say about a graph with loops on a simple graph with loops.

Now we correlate simple graph and graph. Let $G = (V, E)$ be a simple graph. Define $\phi : E \to E$ to be identity map; *i.e.* $\phi(e) = e$ for all $e \in E$.

The graph $G' = (V, E, \phi)$ is essentially the same as G. There is one subtle difference in the picture: The edges of G are unlabeled but each edge of G' is labeled by a set consisting of the two vertices at its ends. But this extra information is contained already in the specification of G. Thus, we can say that simple graphs are special case of graphs.

10.2.4 Degree of Vertices

Let $G = (V, E, \phi)$ be a graph and $v \in V$ a vertex. Define the degree of v, $d(v)$ to be the number of $e \in E$ such that $v \in \phi(e)$; *i.e; e* is incident on v suppose $|v| = n$. Let $d_1, d_2, ... d_n$, where $d_1 \leq d_2 \leq .... \leq d_n$ be the sequence of degrees of the vertices of G, sorted by size. We refer to this sequence as the degree sequence of the graph G.

From Fig. 10.4 (the graph for routes between cities) $d(A) = 3$, $d(B) = 6$, $d(C) = 4$ and $d(D) = 1$. The degree sequence is 1, 3, 4, 6.

Theorem 1: The number of vertices of odd degree in a graph is always even.

Proof: We consider the vertices with odd and even degrees separately, the quantity in the left side of equ.(i) can be expressed as the sum of two sums, each taken ones vertices of even and odd degrees, respectively, as follows:

$$\sum_{i=1}^{n} d\left(v_i\right) = \sum_{\text{even}} d\left(v_j\right) + \sum_{\text{odd}} d\left(v_k\right) \qquad ...(ii)$$

Since left-hand side of equ. (*ii*) is even, and the first expression on the right-hand side is even (being the sum of even numbers), the second expression must also be even:

$$\sum_{\text{odd}} d\left(v_k\right) = \text{an even number} \qquad ...(iii)$$

Since in equ. (*iii*) each $d(v_k)$ is odd, the total number of terms in the sum must be even to make the sum of an even number.

Note: A graph in which all vertices are of equal degree is called a regular graph.

In order to have a partition of S, we must have

(a) the $B(s)$ are nonempty and every $t \in S$ is in some $B(s)$ and

(b) for every $p, q \in S$, $B(p)$ and $B(q)$ are either equal or disjoint.

Since $\sim$ is reflexive, $s \in B(s)$, proving (a). Suppose $x \in B(p) \cap B(q)$ and $y \in B(p)$. We have, $p \sim x$, $q \sim x$ and $p \sim y$. Thus $q \sim x \sim p \sim y$ and so $y \in B(q)$, proving that $B(p) \subseteq B(q)$. Similarly $B(q) \subseteq B(p)$ and so $B(p) = B(q)$. This proves (b). ■

Example 3

Sex in America (Matching Problem)

A 1994 University of Chicago study entitled ***The social Organization of Sexuality*** found that on average men have 74% more opposite-gender partners than women.

Let's recast this observation in graph theoretic terms. Let $G = (V, E)$ be a graph where the set of vertices V consists of everyone in America. Now each vertex either represents either a man or a woman, so we can partition V into two subsets: M, which contains all the male vertices, and W, which contains all the female vertices. Let's draw all the M vertices on the left and the W vertices on the right as shown in Fig. 10.5.

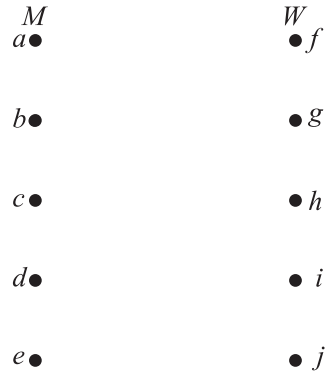


Fig. 10.5: Matching Problem

Now, without getting into a lot of specifics, sometimes an edge appears between an *M* vertex and a *W* vertex. (Fig. 10.6)

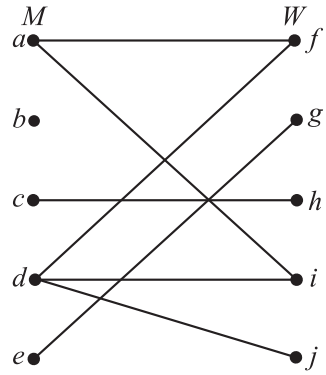


Fig. 10.6: Perfect Match

Since we're only considering opposite-gender relationships, every edge connects an *M* vertex on the left to a *W* vertex on the right. So the sum of degrees of the *M* vertices must equal the sum of the degrees of the *W* vertices:

$$\sum_{x \in M} \deg(x) = \sum_{y \in W} \deg(y)$$

Now suppose we divide both sides of this equation by the product of the sizes of the two sets, $|M|.|W|$:

$$\left(\frac{\sum_{y \in M} \deg(y)}{|M|} \right) \cdot \frac{1}{|W|} = \left(\frac{\sum_{y \in M} \deg(y)}{|M|} \right) \cdot \frac{1}{|W|}$$

The terms above in parentheses are the average degree of an M vertex and the average degree of a W vertex. So we know:

$$\frac{\text{Avg. deg in } M}{|W|} = \frac{\text{Avg. deg in } W}{|M|}$$

$$\text{Avg. deg in M} = \frac{|W|}{|M|} \text{. Avg. deg in } W.$$

Now the Census Bureau reports that there are slightly more women than men in America; in particular $|W| / |M|$ is about 1.035. So– assuming the Census Bureau is correct– we have just proved that the University of Chicago study got bad data! On average, men have 3.5% more opposite-gender partners. Furthermore, this is totally unaffected by differences in sexual practices between men and women; rather, it is completely determined by the relative number of men and women!

We consider a graph G with e edges and n vertices $v_1, v_2,.....v_n$. Since each edge contributes two degrees, the sum of the degrees of all vertices in G is twice the number of edges in G. That is

$$\sum_{i=1}^{n} d(v_i) = 2e \qquad ...(i)$$

From Fig. 10.4. $d(A) + d(B) + d(C) + d(D) = 3 + 6 + 4 + 1 = 14$ *i.e.* twice the number of edges. ■

Example 4

Illustration of Equivalence Relations

Let S be any set and let all the blocks of the partition have one element. Two elements of S are equivalent if and only if they are the same. This rather trivial equivalence relation is, of course, denoted by "=".

Now let the set be the integers Z. Let's try to define an equivalence relation by saying that n and k are equivalent if and only if they differ by a multiple of 24. Is this an equivalence relation? If it is, we should be able to find the blocks of the partition. There are 24 of them, which we call number 0, ..., 23. Block j consists of all integers which equals j plus a multiple of 24; that is, they have a remainder of j when divided by 24. Since two numbers belong to the same block if and only if they both have the same remainder when divided by 24, it follows that they belong to

the same block if and only if their difference gives a remainder of 0 when divided by 24, which is the same as saying their difference is a multiple of 24. Thus this partition does indeed give the desired equivalence relation.

Now let the set be $Z \times Z^*$. where Z^* is the set of all integers except O. Write $(a, b) \sim (c, d)$ if and only if $ad = bc$. With a moment's reflection, you should see that this is a way to check if the two fractions a/b and c/d are equal. We can label each equivalence class with the fraction a/b that it represents. In an axiomatic development of the rationals from the integers, we defines a rational number to be just such an equivalence class and proves that it is possible to add, subtract, multiply and divide equivalence classes.

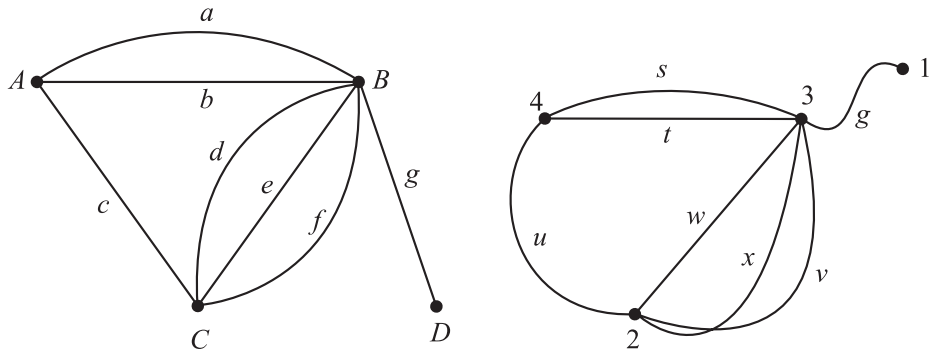
Suppose we consider all functions $S = \underline{m}^{\underline{n}}$.We can define a partition of S in a number of different ways. For example, we could partition the functions f into blocks where the sum of the integers in the Image (f) is constant, where the max of the integers in Image (f) is constant, or where the "type vector" of the function, namely, the number of 1's, 2's, etc. in Image(f), is constant. Each of these defines a partition of S. ■

■ Example 5: *Ghosts of Departed Graphs*

We consider the following two graphs represented by Fig 10.7.

Let us remove all symbols representing edges and vertices. We have left two "forms" on which the graphs were drawn, we can think of drawing a picture of a graph as a two step process

(*i*) draw the form (*ii*) add the labels



G_a G_b

Fig. 10.7: Ghosts of Departed Graphs

We get form F_a and Form F_b have a certain similarity as shown in Fig 1.8.

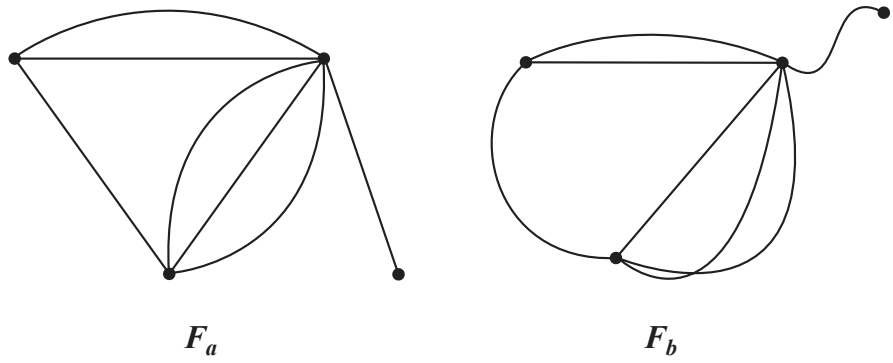


Fig. 10.8: Similarity of Graphs

Now suppose we have any two graphs, $G_a = (V_a, E_a, \phi_b)$. We can think of these graphs not as pictures, but as specified in terms of sets and functions. Now we choose forms F_a and F_b for G_a and G_b respectively, and draw their pictures. We leave it to our intuition to accept the fact that either $F_a \sim F_b$, no matter what we choose for F_a and F_b, or $F_a \not\sim F_b$ no matter what our choice is for the forms F_a and F_b. If $F_a \sim F_b$ we say that G_a and G_b are isomorphic graphs and write $G_a \approx G_b$. The fact that $\sim$ is an equivalence relation forces $\approx$ to be an equivalence relation also. In particular, two graphs G_a and use that same form for G_b. ■

10.2.5 Equivalence Relation

An equivalence relation on a set S is a partition of S. We say that $s, t \in S$ are equivalent if and only if they belong to the same block (called an equivalence class in this context) of the partition. If the symbol $\sim$ denotes the equivalence relation, then we write $s \sim t$ to indicate that s and t are equivalent.

Theorem 2: Let S be a set and suppose that we have a binary relation $R \subseteq S \times S$. We write $s \sim t$ whenever $(s, t) \in R$. This is an equivalence relation if and only if the following three conditions hold.

(*i*) (Reflexive) For all $s \in S$ we have $s \sim S$.

(*ii*) (Symmetric) For all $s, t \in S$ such that $s \sim t$ we have $t \sim s$.

(*iii*) (Transitive) For all $r, s, t \in S$ such that $r \sim s$ and $s \sim t$ we have $r \sim t$.

Proof: We first prove that an equivalence relation satisfies (*i*)-(*iii*). Suppose that $\sim$ is an equivalence relation. Since s belongs to whatever block it is in, we have $s \sim s$. Since $s \sim t$ means that s and t belong to the

same block, we have $s \sim t$ if and only if we have $t \sim s$. Now suppose that $r \sim s \sim t$. Then r and s are in the same block and s and t are in the same block. Thus r and t are in the same block and so $r \sim t$.

We now suppose that (*i*)-(*iii*) hold and prove that we have an equivalence relation. What would the blocks of the partition be? Everything equivalent to a given element should be in the same block. Thus, for each $s \in S$ let $B(s)$ be the set of all $t \in S$ such that $s \sim t$. We must show that the set of these sets from a partition of S. ■

10.2.6 Random Graph Model

Let $G(n, p)$ be the probability space obtained by letting the elementary events be the set of all n-vertex simple graphs with $V = \underline{n}$. If $G \in G(n, p)$ has m edges, the $P(G) = p^m q^{N-m}$ where $q = 1 - p$ and $N = \binom{n}{2}$

We need to show that $G(n, p)$ is a probability space. There is a nice way to see this by reinterpreting P. List the $N = \binom{n}{2}$ vertices $P_2(V)$ in lex order. Let the sample space be $U = x^N$ {choose, reject} with $P(a_1,...,a_1,...,a_N) = P^*(a_1) \times ... \times P^*(a_N)$ where P^* (choose) $= p$ and P^* (reject) $= 1 - p$. We've met this before in Unit F_n and seen that it is a probability space. To see that it is, note that $P \geq 0$ and

$$\sum_{a_1,...,a_N} P(a_1,...,a_N) = \sum_{a_1,...,a_N} P^*(a_1) \times \cdots \times P^*(a_N)$$

$$= \left(\sum_{a_1} P^*(a_1)\right) \times \cdots \times \left(\sum_{a_N} P^*(a_N)\right)$$

$$= (p + (1 - p)) \times ... \times (p + (1 - p))$$

$$= 1 \times ... \times 1 = 1.$$

We think of the chosen pairs as the edges of a graph chosen randomly from $G(n, p)$. If G has m edges, then its probability should be $p^m(1 - p)^{N-m}$ according to the definition. On the other hand, since G has m edges, exactly m of $a_1, ..., a_N$ equal "choose" and so, in the new space, $P(a_1,...,a_N) = p^m (1 - P)^{N-m}$ also. We say that we are choosing the edges of the random graph independently.

10.2.7 Isolated Vertex, Pendent Vertex and Null Graph

A vertex having no incident edge is called an isolated vertex or isolated vertices are the vertices with zero degree.

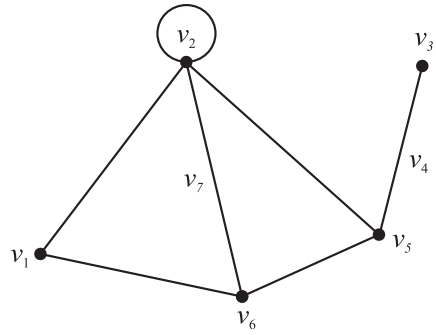


Fig. 10.9: Isolated Vertex

In Fig. 10.9 vertices v_4 and v_7 are isolated vertices.

A vertex of degree one is called a pendent vertex or an end vertex. Vertex in Fig. 10.9 is a pendent vertex. Two adjacent edges are said to be in series if their common vertex is of degree two. In Fig. 10.9, the two edges incident on v_1 are in series.

Note: A graph without any edges, is called a null graph.

10.3 Digraphs

A directed graph (or digraph) is a triple $D = (V, E, \phi)$ where V and E are finite sets and ϕ is a function with domain E and codomain $V \times V$. We call E the set of edges of the digraph D and call V the set of vertices of D.

We can define a notion of a simple digraph. A simple digraph is a pair $D = (V, E)$, where V is a set, the vertex set, and $E\, x \subseteq V \times V$ is the edge set. Just as with simple graphs and graphs, simple digraphs are a special case of digraphs in which ϕ is the identify function on E; *i.e.* $Q(e) = e \ ^{\forall} \ e \in E$.

There is a correspondence between simple graphs and simple digraphs that is fairly common in application of graph theory. To interpret simple graphs in terms of simple digraphs, it is best to consider simple graphs with loops.

We consider $G = (V, E)$ where $E \subseteq P_2 (V) \cup P_1(V)$. We may identify $\{u, v\} \in P_2(V) \cup P_1(V)$ with $(u, v) \in V \times V$ and with $(v, u) \in V \times V$. We identify $\{4\}$ with (u, v). The Fig. 10.10 shows the simple graph and corresponding digraph.

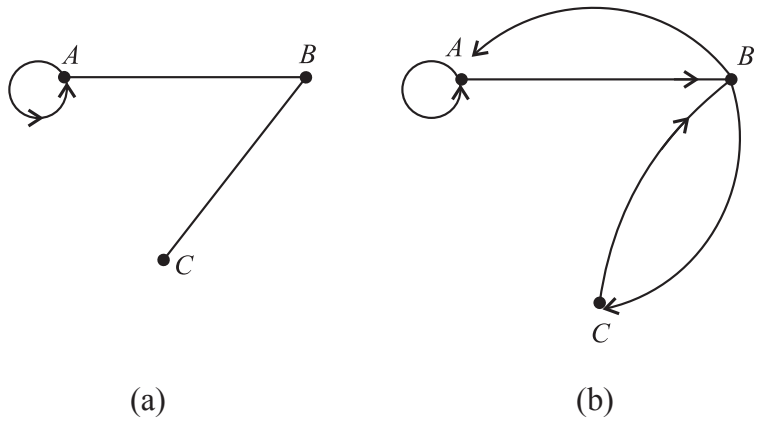


Fig. 10.10: Show the Simple Graph and Corresponding Digraph

Each, edge that is not a loop in the simple graph is replaced by two edges "in opposite directions" in the corresponding simple digraph. A loop is replaced by a directed loop (*e.g.*, $\{A\}$ is replaced by (A, A)).

Simple digraphs appear in mathematics under another important guise: binary relations. A binary relation on a set V is simply a subset of $V \times V$. Often the name of the relation and the subset are the same. Thus we speak of the binary relation $E \subseteq V \times V$. If you have absorbed all the terminology, you should be able to see immediately that (V, E) is a simple digraph and that any simple digraph $(V' \times V')$ correspondence to a binary relation $E' \subseteq V' \times V'$.

We can recall that a binary relation R is called symmetric if $(u, v) \in R$ implies $(v, u) \in R$. Thus simple graphs with loops correspond to symmetric binary relations.

An equivalence relation on a set S is a particular type of binary relation $R \subseteq S \times S$. For an equivalence relation, we have $(x, y) \in R$ if and only if x and y are equivalent (*i.e.*, belong to the same equivalence class or block). Note that this is a symmetric relationship, so we may regard the associated simple digraph as a simple graph. Which simple graphs (with loops allowed) correspond to equivalence relations? As an example, take $S = \underline{7}$ and take the equivalence class partition to be $\{\{1, 2, 3, 4\}, \{5, 6, 7\}\}$. Since everything in each block is related to everything else, there are $\binom{4}{2} = 6$ non-loops and $\binom{4}{2} = 4$ loops associated with the block $\{1, 2, 3, 4\}$ for a total of ten edges. With the block $\{5, 6, 7\}$ there are three loops and three non-loops for a total of six edges. Here is the graph of this equivalence relation: (Fig. 10.11)

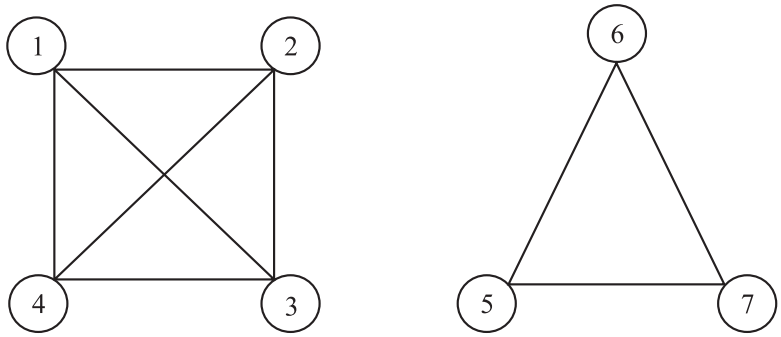


Fig. 10.11: Graph of Equivalence Relation

A complete simple graph $G = (V, E)$ with loops is a graph with every possible edge. That is, $E = P_2(V) \cup P_1(V)$. In the above graph, each block of the equivalence relation is replaced by the complete simple graph with loops on that block.

Handshaking Lemma: For a graph G we have

$$\sum_{u \in V(G)} d_G(u) = 2 \mid E(G) \mid$$

Example 6

Flow of Commodities Problem

We can recall example 2. Imagine now that the symbols a, b, c, d, e, f and g, instead of standing for route names, stand for commodities (applesauce, bread, computers, etc.) that are produced in one town and shipped to another town. In order to get a picture of the flow of commodities, we need to know the directions in which they are shipped. This information is provided by picture below (Fig. 10.12)

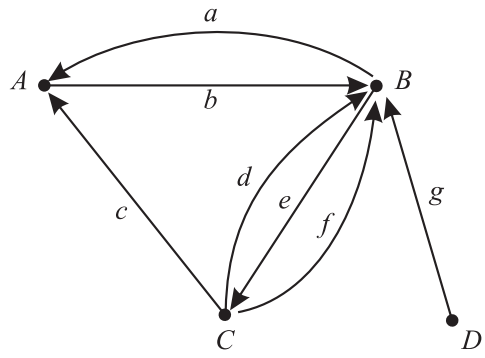


Fig. 10.12: Flow of Commodities Problem

In set-theoretic terms, the information needed to construct the above picture can be specified by giving a pair $D = (V, E, \phi)$ where ϕ is a function. The domain of the function ϕ is $E = \{a, b, c, d, e, f, g\}$ and the codomain is $V \times V$. Specifically,

$$\phi = \begin{pmatrix} a & b & c & d & e & f & g \\ (B,A) & (A,B) & (C,A) & (C,B) & (B,C) & (C,B) & (D,B) \end{pmatrix} \quad \blacksquare$$

10.4 Path, Trail, Walk and Vertex Sequence

Let $G = (V, E, \phi)$ be a graph. Let $e_1, e_2,...,e_{n-1}$ be a sequence of elements of E (edges of G) for which there is a sequence $a_1, a_2,...,a_n$ of distinct elements of V (vertices of G) such that $\phi(e_i) = \{a_i, a_{i+1}\}$ for $i = 1, 2,..., n - 1$. The sequence of edges $e_1, e_2,...,e_{n-1}$ is called a path in G. The sequence of vertices $a_1, a_2,...,a_n$ is called the vertex sequence of the path. (Note that since the vertices are distinct, so are the edges.)

If we require that $e_1,...,e_{n-1}$ be distinct, but not that $a_1,...,a_n$ be distinct, the sequence of edges is called a *trail*.

If we do not even require that the edges be distinct, it is called a *walk*.

If $G = (V, E, \phi)$ is a directed graph, then $\phi(e_i) = \{a_i, a_{i+1}\}$ is replaced by $\phi(e_i) = (a_i, a_{i+1})$ in the above definition to obtain a directed path, trail, and walk respectively.

Note: *Every path is a trail and every trail is a walk, but not conversely.*

Theorem 3: Walk implies path

Suppose $u \neq v$ are the vertices is a graph $G = (V, E, \phi)$.

The following are equivalent:

(*i*) There is a walk from u to v.

(*ii*) There is a trail from u to v.

(*iii*) There is a path from u to v.

Furthermore, given a walk from u to v, there is a path from u to v all of whose edges are in the walk.

Proof: Since every path is a trail, (*iii*) implies (*ii*). Since every trail is a walk, (*ii*) implies (*i*). Thus it suffices to prove that (*i*) implies (*iii*). Let $e_1, e_2, ...e_k$ be a walk from u to v. We use induction on n, the number of repeated vertices in a walk. If the walk has no repeated vertices, it is a path. This starts the induction at $n = 0$. Suppose $n > 0$. Let r be a repeated vertex. Suppose it first appears in edge e_i and last appears in edge e_j.

If $r = u$, then $e_j,...,e_k$ is a walk from u to v in which r is not a repeated vertex. If $r = v$, then $e_1,...,e_i$ is a walk from u to v in which r is not a repeated vertex. Otherwise, $e_1,...,e_i, e_j,...,e_k$ is a walk from u to v in which r is not a repeated vertex. Hence there are less than n repeated vertices in this walk from u to v and so there is a path by induction. Since we constructed the path by removing edges from the walk, the last statement in the theorem follows. ■

10.5 Subgraph

Let $G = (V, E, \phi)$ be a graph. A graph $G' = (V', E', \phi')$ is a subgraph of G iff $V' \subseteq V$, $E' \subseteq E$, and ϕ' is the restriction of ϕ to E'.

The fact that G' is itself a graph means that $\phi(x) \in P_2(V')$ for each $x \in E'$ and, in fact, the codomain of ϕ' must be $P_2(V')$. If G is a graph with loops, the codomain of ϕ' must be $P_2(V') \cup P_1(V')$. This definition works equally well if G is a digraph. In that case, the codomain of ϕ' must be $V \times V$.

10.6 Circuit and Cycle

Let $G = (V, E, \phi)$ be a graph and let $e_1, e_2,...,e_n$ be a trail with vertex sequence $a_1, a_2,...,a_n, a_1$. (returns to the starting point) The subgraph G' of G induced by the set of edges $\{e_1, e_2,...,e_n\}$ is called a circuit of G. The length of the circuit is n.

- If the only repeated vertices on the trail are a, (the start and the end), then the circuit is called a simple circuit or cycle.
- If 'trail' is replaced by directed trail, we obtain a directed circuit and a directed cycle.

Note: *In the above definitions, a path is a sequence of edges but a cycle is a subgraph of G. In actual practice, we often think of a cycle as a path, except that it starts and ends at the same vertex. This sloppiness rarely causes trouble, but can lead to problems in formal proofs. Cycles are closely related to the existence of multiple paths between vertices.*

Example 7

Subgraph-Key Information

For the graph $G = (V, E, \phi)$ as shown in Fig. 10.13, let $G' = (V', E', \phi')$ be defined by $V' = \{A, B, C\}$, $E' = \{a, b, c, f\}$, and by ϕ' being the

restriction of f to E' with codomain $P_2(V')$. ϕ' is determined completely from knowing V', E' and ϕ. Thus, to specify a subgraph G', the key information is V' and E'.

In the same graph, let $V' = V$ and $E' = \{a, b, c, f\}$. In this case, the vertex D is not a member of any edge of the subgraph. Such a vertex is called an isolated vertex of G'.

We may specify a subgraph by giving a set of edges $E' \subseteq E$ and taking V' to be the set of all vertices on some edge of E'.

It shows that, V', is the union of the sets $\phi(x)$ over all $x \in E'$. Such a subgraph is called the subgraph induced by the edge set E' or the edge induced subgraph of E'. The first subgraph of this example is the subgraph induced by $E' = \{a, b, c, f\}$.

Given a set $V' \subseteq V$, we can take E' to be the set of all edges $x \in E$ such that $\phi(x) \subseteq V'$. The resulting subgraph is called the subgraph induced by V' or the vertex induced subgraph of V'. In the given Fig., the edges of the subgraph induced by $V' = \{C, B\}$, are $E' = \{d, e, f\}$

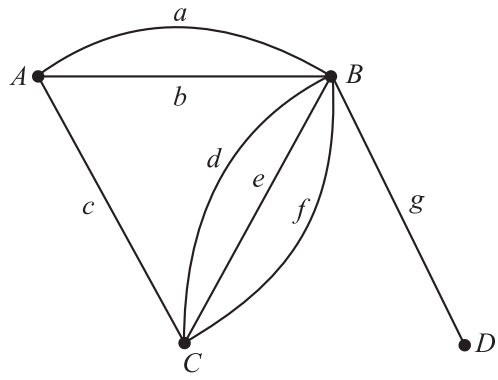


Fig. 10.13: Parallel Edges

We consider the path c, a with vertex sequence C, A, B. The edge d has $\phi(d) = \{C, B\}$. The subgraph $G' = (V', E', \phi')$ where $V' = \{C, A, B\}$ and $E' = \{c, a, d\}$ is called a cycle of G. Whenever there is a path in G, (say $e_1, e_2,...,e_{n-1}$ with vertex sequence $a_1, a_2,...,a_n$) and an edge x with $\phi(x) = \{a_1, a_n\}$, then the subgraph induced by the edges $e_1, e_2,...,e_{n-1}$ x is called the cycle of G.

Parallel edges like a and b in the Fig. 10.13 induce a cycle. A loop also induces a cycle. ■

10.7 Cycles and Multiple Paths

Two vertices $u \neq v$ are on a cycle of graph G iff there are at least two paths from u to v that have no vertices in common except the end points u and v.

To illustrate the above, we consider *u* and *v* are on a cycle. We follow the cycle from *u* to *v* to obtain one path. Then we follow the cycle from *v* to *u* to obtain another. Since the cycle has no repeated vertices, the only vertices that lie on both paths are *u* and *v*. On the other hand, a path from *u* to *v* followed by a path from *v* to *u* is a cycle if the paths have no vertices in common other than *u* and *v*.

10.8 Connected Graph

Let $G = (V, E, \phi)$ be a graph. If for any two distinct elements u and v of V there is a path P from u to v than G is a connected graph. If $| V | = l$, then G is connected.

To illustrate this definition we refer example 1 which has two distinct pieces. It is not a connected graph. There is, for example, no path from $u = NB$ to $v = AC$. Note that one piece of this graph consists of the vertex induced subgraph of the vertex set $\{AC, SK, BM, RK\}$ and the other piece consists of the vertex induced subgraph of $\{NB, AR, SJ, HB\}$. These pieces are called connected components of the graph. This is the case in general for a graph $G = (V, E, \phi)$.

The vertex set is partitioned into subsets $V_1, V_2,...,V_m$ such that if *u* and *v* are in the same subset then there is a path from *u* to *v* and if they are in different subsets there is no such path. The subgraphs $G_1 = (V_1, E_1, \phi_1),...,G_m = (V_m, E_m, \phi_m)$ induced by the sets $V_1, V_2,...,V_m$ are called the connected components of *G*. Every edge of *G* appears in one of the connected components. We can see this by considering that $\{u, v\}$ is an edge and note that the edge is a path from *u* to *v* and so *u* and *v* are in the same induced subgraph, G_1.

Example 8

Connected Components as an Equivalence Relation

We have to define firstly, two integers to be 'connected' if they have a common factor. Let 2 and 6 are connected and 3 and 6 are connected, but 2 and 3 are not connected and so we cannot partition the set $V = \{2, 3, 6\}$ into 'connected components'.

We must use some property of definitions of graphs and paths to show that the partitioning of vertices is possible. We can use to do this by constructing an equivalence relation.

For $u, v \in V$, we write $u \sim v$ iff either $u = v$ or there is a walk from u to v. It is clear that $\sim$ is reflexive and symmetric. We can prove that it is transitive also. Let $u \sim v \sim w$. The walk from u to v followed by the walk from v to w is a walk from u to w. This completes the proof that $u \sim v$ is an equivalent relation. The relation partitions V onto subsets V_1, V_2,...,V_m. The vertex induced subgraphs of the V_i satisfy the definition of connected graph. ■

10.9 Spanning Subgraph and Induced Subgraph

A spanning subgraph is a subgraph containing all the vertices of G.

i.e. if $V(H) \subset V(G)$ and $E(H) \subseteq E(G)$ then H is a proper subgraph of G and if $V(H) = V(G)$ then it can be said that H is a spanning subgraph of G.

A spanning subgraph need not contain all the edges in G.

For any set S of vertices of G, the vertex induced subgraph or simply an induced subgraph $< S >$ is the maximal subgraph of G with vertex set S. Thus two vertices of S are adjacent in $< S >$ if and only if they are adjacent in G.

In other words, if G is a graph with vertex set V and U is a subset of V then the subgraph $G(U)$ of G whose vertex set is U and whose edge set comprises exactly the edge of E which join vertices in U is termed as induced subgraph of G.

10.10 Eulerian Graph (Eulerian Trail and Circuit)

Let $G = (V, E, \phi)$ be a connected graph. If there is a trail with edge sequence $(e_1, e_2,...,e_k)$ in G, which uses each edge in E, then $(e_1, e_2,...,e_k)$ is called the Eulerian trail. If there is a circuit $C = (V', E', \phi')$ in G with $E' = E$, then C is called an Eulerian circuit.

We can describe a process for constructing a graph $G = (V, E, \phi)$.

Start with $V = \{v_1\}$ consisting of a single vertex and with $E = \phi$. Add an edge e_1, with $\phi(e_1) = \{v_1, v_2\}$, to E. If $v_1 = v_2$, we have a graph with one vertex and one edge (a loop), else we have a graph with two vertices and one edge. Keep track of the vertices and edges in the order added. Here (v_1, v_2) is the sequence of vertices in the order added and

(e_1) is the sequence of edges in order added. Suppose we continue this process to construct a sequence of vertices (not necessarily distinct) and sequence of distinct edges. At the point where k distinct edges have been added, if v is the last vertex added, then we add a new edge e_{k+1}, different from all previous edges, with $\phi(e_{k+1}) = \{v, v'\}$ where either v' is a vertex already added or a new vertex. Here is a picture of this process carried out with the edges numbered in the order added (Fig. 10.14)

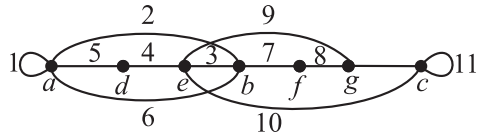


Fig. 10.14: Eulerian Graph

where the vertex sequence is

$$S = (a, a, b, e, d, a, b, f, g, e, c, c, g).$$

Such a graph is called a graph with an *Eulerian trail*. The edges, in the order added, are the *Eulerian trail* and S is the vertex sequence of the trail.

By construction, if G is a graph with an eulerian trail, then there is a trail in G that includes every edge in G. If there is a circuit in G that includes every edge of G then G is called an Eulerian circuit graph or graph with an Eulerian circuit. Thinking about the above example, if a graph has an Eulerian trail but no Eulerian circuit, then all vertices of the graph have even degree except the start vertex (a in our example with degree 5) and end vertex (g in our example with degree 3). If a graph has an Eulerian circuit then all vertices have even degree. The converses in each case are also true (but take a little work to show): If G is a connected graph in which every vertex has even degree then G has an Eulerian circuit. If G is a connected graph with all vertices but two of even degree, then G has an Eulerian trail joining the two vertices of odd degree.

10.11 Hamiltonian Graph

A cycle is a graph G = (V, E, ϕ) is a Hamiltonian cycle for G is every element of v is a vertex of the cycle. A graph G = (V, E, ϕ) is Hamiltonian if it has a subgraph that is a Hamiltonian cycle for G.

We can start with a graph $G' = (V, E', \phi')$ that is a cycle and then add additional edges, without adding any new vertices, to obtain a graph $G = (V, E, \phi)$. As an example, consider (Fig. 10.15)

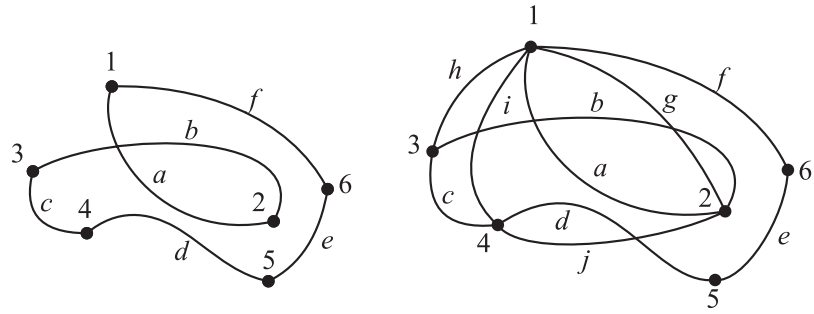


Fig. 10.15: Hamiltonian Graph

(V, E', ϕ') is the cycle induced by the edges $\{a, b, c, d, e, f\}$. The second graph $G = (V, E, \phi)$ is obtained from G' by adding edges g, h, i and j. A graph that can be constructed from such a two-step process is called a Hamiltonian graph. The cycle G' is called a Hamiltonian cycle of G.

10.12 Biconnected Graph

Let $G = (V, E, \phi)$ be a graph. For $e, f \in E$ write $e \sim f$ if either $e = f$ or there is a cycle of G that contains both e and f. We claim that this is an equivalence relation. The reflexive and symmetric parts are easy. Suppose that $e \sim f \sim g$. If $e = g$, then $e \sim g$, so suppose that $e \neq g$. Let $\phi(e) = \{v_1, v_2\}$. Let $C(e, f)$ be the cycle containing e and f and $C(f, g)$ the cycle containing f and g. In $C(e, f)$ there is a path P_1 from v_1 to v_2 that does not contain e. Let x and $y \neq x$ be the first and last vertices on P_1 that lie on the cycle containing f and g. We know that there must be such points because the edge f is on P_1. Let P_2 be the path in $C(e, f)$ from y to x containing e. In $C(f, g)$ there is a path P_3 from x to y containing g. We claim that P_2 followed by P_3 defines a cycle containing e and g.

Since $\sim$ is an equivalence relation on the edges of G, it partitions them. If the partition has only one block, then we say that G is a biconnected graph. If E' is a block in the partition, the subgraph of G induced by E' is called a bicomponent of G. Note that the bicomponents of G are not necessarily disjoint: Bicomponents may have vertices in common (but never edges). There are four bicomponents in the following graph. (Fig. 10.16) Two are the cycles, one is the edge $\{C, O\}$, and the fourth consists of all of the rest of the edges.

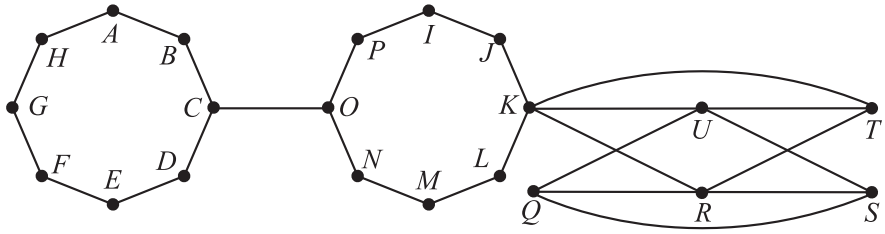


Fig. 10.16: Biconnected Graph

10.13 Algebraic terms and operations used in Graph Theory

10.13.1 Graphs Isomorphism

Let $G_1 = (v_1, E_1)$ and $G_2 = (v_2, E_2)$ be two graphs. A function $f : u_1 \rightarrow v_1$ is called a graphs isomorphism iff.

(*i*) *f is one-to-one and onto.*

(*ii*) *for all a, b $\in v_1$, {a, b} $\in E_1$ iff {f(a), f(b) $\in E_2$ when such a function exists, G_1, and G_2 are called isomorphic graphs and written as $G_1 \cong G_2$.*

OR

Two graphs G_1 and G_2 are called isomorphic to each other if there is a one-to-one correspondence between their vertices and between edges such that incidence relationship is preserve. Written as $G_1 \cong G_2$ or $G_1 = G_2$.

The necessary conditions for two graphs to be isomorphic are:

(*i*) Both graphs must have the same number of vertices.

(*ii*) Both graphs must have the same number of edges.

(*iii*) Both graphs must have equal number of vertices with the same degree,

(*iv*) They must have the same degree sequence and same cycle vector $(c_1, c_2,...,c_n)$, where e_i is the number of cycles of length *i*.

e.g. Fig. 10.17 has isomorphism of graphs.

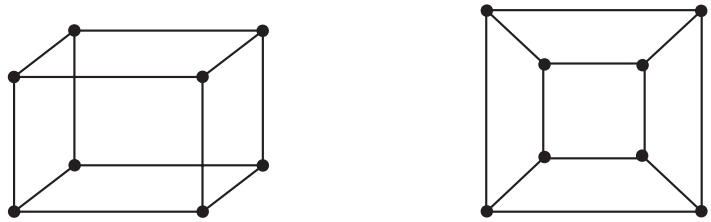
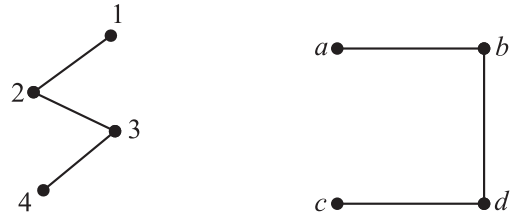


Fig. 10.17: Isomorphism of Graphs

Example 9

Show that the two Graphs shown in Fig. 10.18 are Isomorphic



(G_1) (G_2)

Fig. 10.18

Solution:

Observing the above graphs G_1 and G_2, we find.

$$V(G_1) = \{1, 2, 3, 4\},\ V(G_2) = \{a, b, c, d\}$$

$$E(G_1) = \{\{1, 2\}, \{2, 3\}, \{3, 4)\}$$

and $$E(G_2) = \{\{a, b\}, \{b, d\}, \{d, c\}\}.$$

Defining a function $f: V(G_1) \to V(G_2)$ as.

$$f(1) = a, f(2) = b, f(3) = d, \text{ and } f(4) = c.$$

f is clearly one-one and onto, hence an isomorphism.

Furthermore, $\{1, 2\} \in E(G_1)$ and $\{f(1), f(2)\} = \{a, a\} \in E(G_2)$

$$\{2, 3\} \in E(G_1) \text{ and } \{f(2), f(3)\} = \{b, d\} \in E(G_2)$$

$$\{3, 4\} \in E(G_1) \text{ and } \{f(3), f(4)\} = \{d, c\} \in E(G_2)$$

and $$\{1, 2\} \notin E(G_1) \text{ and } \{f(1), f(3)\} = \{a, d\} \notin E(G_2)$$

$$\{1, 4\} \notin E(G_1) \text{ and } \{f(1), f(4)\} = \{a, c\} \notin E(G_2)$$

$$\{2, 4\} \notin E(G_1) \text{ and } \{f(2), f(4)\} = \{b, c\} \notin E(G_2)$$

Hence *f* preserves adjacency as well as non-adjacency of the vertices. Therefore, G_1 and G_2 are isomorphic graphs. ■

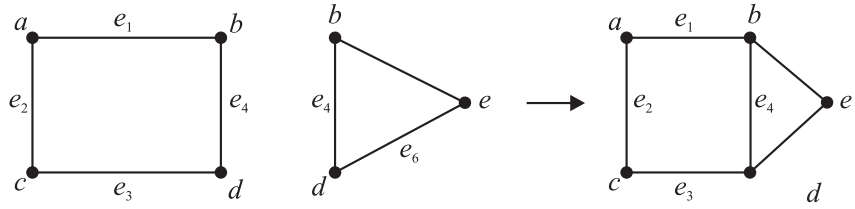
10.13.2 Union of two Graphs

If we have two graphs G_1 and G_2, then, their union will be graph such that

$$V(G_1 \cup G_2) = V(G_1) \cup V(G_2)$$

and $$E(G_1 \cup G_2) = E(G_1) \cup E(G_2)$$

Fig. 10.19 Shows the union operation of two graphs G_1 and G_2.



(G_1) (G_2) $(G_1 \cup G_2)$

Fig. 10.19: Union of two Graphs

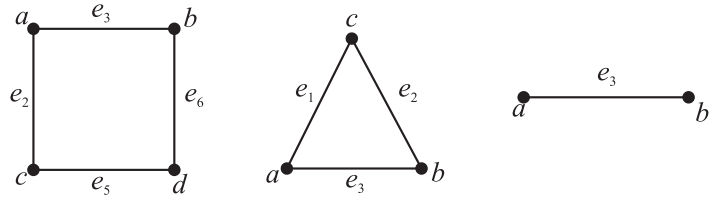
10.13.3 Intersection of two Graphs

If we have two graphs G_1 and G_2 with at least one vertex is common then their intersection will be a graph such that

$$V(G_1 \cap G_2) = V(G_1) \cap V(G_2)$$

and $$E(G_1 \cap G_2) = E(G_1) \cap E(G_2)$$

Fig. 10.20 illustrates the intersection operation of two graphs G_1 and G_2.



(G_1) (G_2) $(G_1 \cap G_2)$

Fig. 10.20: Intersection of two Graphs

10.13.4 Addition of two Graphs

If we have two graphs G_1 and G_2 such that $V(G_1) \cap V(G_2) = \phi$, then the sum $G_1 + G_2$ is defined as the graph whose vertex set is $V(G_1) + V(G_2)$ and the edge set is consisting these edges, which are G_1 and in G_2 and the edges contained, by joining each vertex of G_1 to each vertex of G_2. Fig. 10.21 illustrates the addition of two graphs G_1 and G_2.

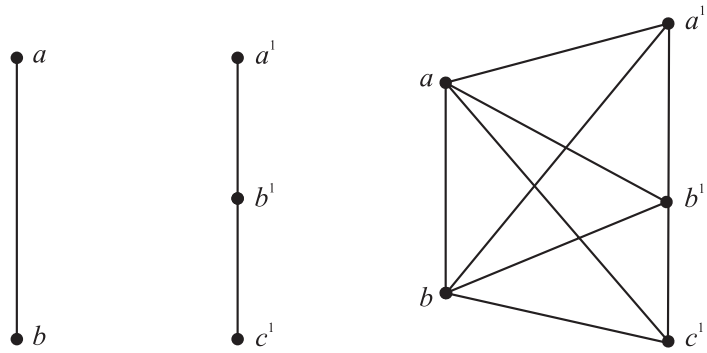


Fig. 10.21: Addition of two Graphs

10.13.5 Direct Sum or Ring Sum of two Graphs

If we have two graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$. Then the direct sum of G_1 and G_2 (denoted by G_1, G_2) is defined as the graph G such that

(*i*) $V(G) = V(G_1) \cup V(G_2)$

(*ii*) $E(G) = E(G_1) \cup E(G_2) - E(G_1) \cap E(G_2)$

i.e. the edges that either in G_1 or G_2 but not in both. The direct sum is illustrated in Fig. 10.22 for two graphs G_1 and G_2.

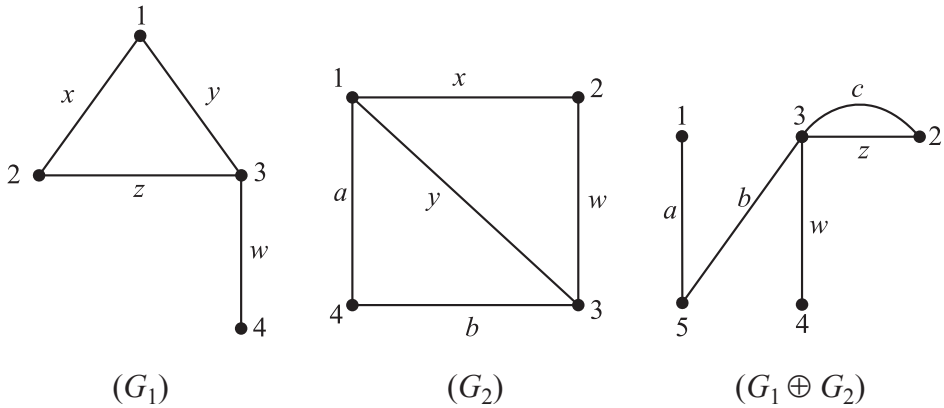


Fig. 10.22: Ring Sum of two Graphs

10.13.6 Product of two Graphs

We can define the product of two graphs $G_1 \times G_2$ by considering any two points $u = (u_1, u_2)$ and $v = (v_1, v_2)$ is $V = V_1 \times V_2$. Then u and v are adjacent in $G_1 \times G_2$ whenever [$u_1 = v_1$ and u_2 adj. v_2] or [v_1 and u_1 adj v_1].

Fig. 10.23 illustrates the product of two graphs ($G_1 \times G_2$).

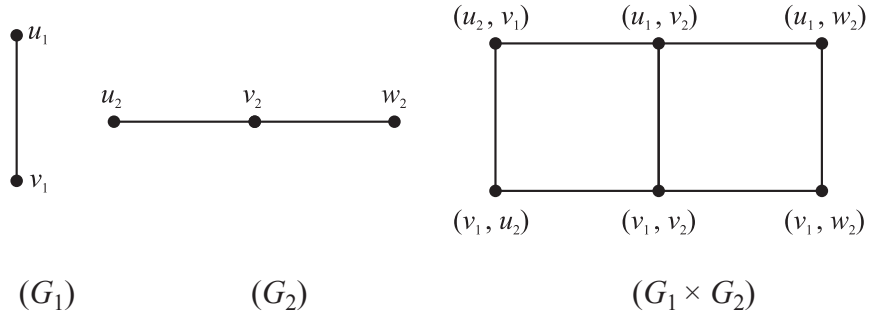


Fig. 10.23: Products of two Graphs

10.13.7 Composition of two Graphs

The composition $G = G_1[G_2]$ also has $v = v_1 \times v_2$ as its point set, and $u = (u_1, u_2)$ is adjacent with $v = (v_1, v_2)$ whenever (u_1 adj. v_1) or ($u_1 = v_1$ and u_2 adj. v_1).

Fig. 10.24 illustrates the composition of two graphs.

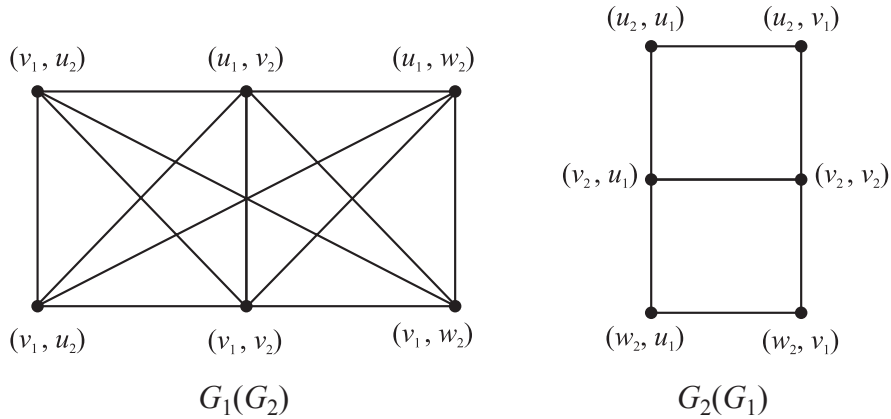
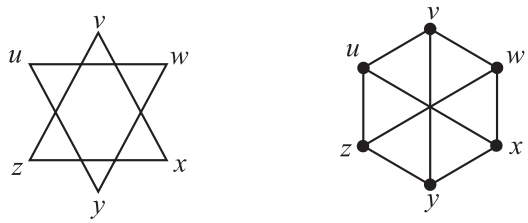


Fig. 10.24: Composition of two Graphs

10.13.8 Complement of a Graph

The complement G' of a graph G may be defined as a simple graph with the same vertex set as G and where two vertices u and v adjacent only when they are not adjacent in G.

Fig. 10.25(*a*) shows the complement of a graph



(G) (G^1)

Fig. 10.25 (a): Complement of a Graph

A graph G is *self-complementary* if it is isomorphic to its complement. Fig. 10.25(*b*) shows this situation.

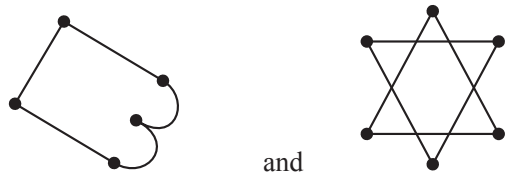


Fig. 10.25 (b): Self Complement of a Graph

The self-complementary graph with five vertices is (as shown in Fig. 10.25(*c*))

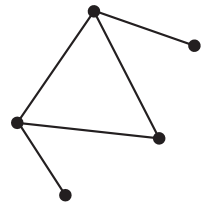


Fig. 10.25 (c): Self Complementary with Five Vertices

10.13.9 Fusion of a Graph

A pair of vertices v_1 and v_2 in a graph G is said to be '*fused*' if these two vertices are replaced by a single new vertex v s.t. every edge that was adjacent to either v_1 or v_2 on both is adjacent v.

We can observe that the fusion of two vertices does not alter the number of edges of graphs but reduced the vertices by one. Fig. 10.26 shows this kind of fusion.

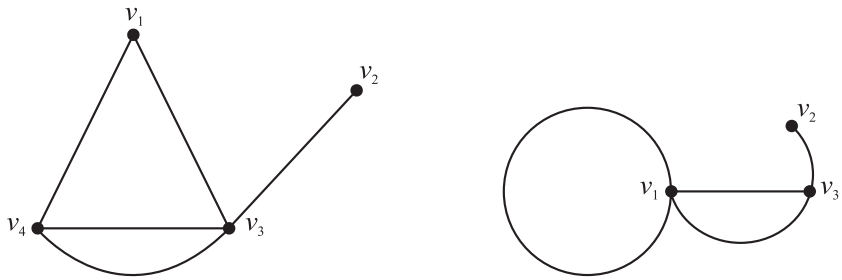


Fig. 10.26: Fusion of Graph

10.13.10 Rank and Nullity

Let G be a graph with n vertices, m edges and k components. We define the rank $P(G)$ and nullity $u(G)$ of G as follows:

$$P(G) = \text{Rank of } G = n - k$$

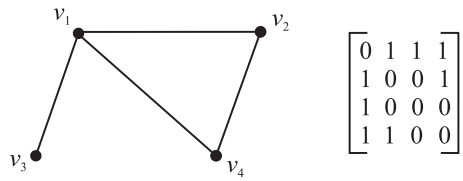
$$\mu(G) = \text{Nullity of } G = m - n + k$$

If G is connected Hence,

$$P(G) = n - 1 \quad \text{and} \quad v(G) = m - n + 1.$$

10.13.11 Adjacency Matrix

A graph can be represented by an adjacency matrix. In particular, if a graph has vertices $v_1,...,v_n$, then the adjacency matrix is $n \times n$. The entry in row i, column j is 1 if there is an edge v_i–v_j and is 0 if there is no such edge. For example, here is a graph and its adjacency matrix:



$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

The adjacency matrix of an undirected graph is always symmetric about the diagonal line running from the upper left entry to the lower right. The adjacency matrix of a directed graph need not be symmetric,

however. Entries on the diagonal of an adjacency matrix are nonzero only if the graph contains self-loops.

Adjacency matrices are useful for two reasons. First, they provide one way to represent a graph in computer memory. Second, by mapping graphs to the world of matrices, one can bring all the machinery of linear algebra to bear on the study of graphs. For example, one can analyze a highly-prized quality of graphs called "expansion" by looking at eigenvalues of the adjacency matrix. (In a graph with good expansion, the number of edges departing each subset of vertices is at least proportional to the size of the subset. This is not so easy to achieve when the graph as a whole as few edges, say $|E| = 3|V|$.) Here we prove a simpler theorem in this vein. If M is a matrix, then M_{ij} denotes the entry in row i, column j. Let M^k denote the k-th power of M. As a special case, M^0 is the identity matrix.

10.13.12 Some Important Theorems

Theorem 3: The maximum number of edges in a simple graph with n vertices is $\frac{n(n-1)}{2}$

Proof: By handshaking theorem, we have

$$\sum_{i=1}^{n} f(v_i) = 2e$$

Where e is the number of edges with vertices in the graph G.

$$d(v_1) + d(v_2) + ... + d(v_n) = 2e. \qquad ...(i)$$

Since we know that maximum degree of each vertex in a graph G can be $(n - 1)$.

Therefore, equ (i) reduces to

$$(n - 1) + (n - 1) +.... \text{ to } n \text{ terms} = 2e$$

$$n(n - 1) = 2e$$

$$e = \frac{n(n-1)}{2}$$

Hence the maximum number of edges in any simple graph with n vertices is $\frac{n(n-1)}{2}$. ■

Theorem 4: *The maximum number of lines among all p point graphs with no triangles is* $\left\lfloor \frac{p^2}{4} \right\rfloor$.

Proof: Suppose the statement is true for all even $p \le 2n$.

We then prove for $p = 2n + 2$.

Let G be a graph with $p = 2n + 2$ points and no triangles.

Since G is not totally disconnected, there are adjacent points u and v.

The subgraph $G' = G - \{u, v\}$ has 2^n points and no triangles, so that by the induction hypothesis.

G' has at most $\left\lfloor \frac{4n^2}{4} \right\rfloor = n^2$ lines.

There might not be a point w *s.t,* u and v are both adjacent to w, for then u, v and w would be points of a triangle in G.

If u is adjacent to k points of G', v can be adjacent to at most $2n - k$ points.

Thus G has at most

$$n^2 + k + (2n - k) + 1 = n^2 + 2n + 1 = \frac{p^2}{4} = \left\lfloor \frac{p^2}{4} \right\rfloor \text{ lines.} \quad \blacksquare$$

Theorem 5: A simple graph with n vertices must be connected if it has more than $\frac{(n-1)(n-2)}{2}$ edges.

Proof: Let G be a simple graph on n vertices.

We can choose $n - 1$ vertices $v_1, v_2, \ldots v_{n-1}$ of G.

We have maximum $\binom{n-1}{2} = \frac{(n-1)(n-2)}{2}$ number of edges.

Only can be drawn between there vertices.

If we have more than $\frac{(n-1)(n-2)}{2}$ edges, at least one edge should be drawn between the n^{th} vertex v_n to some vertex v_i, $1 \le i \le n - 1$ of G.

Hence G must be connected. $\blacksquare$

Theorem 6: A simple graph with n vertices and k components cannot have more than $\frac{(n-1)(n-2)}{2}$ edges. If $m > \frac{1}{2}(n-1)(n-2)$ then a simple graph with n vertices and m edges in connected.

Proof: Let n_i the number of vertices is component i, $i \le i \le k$

Then $$\sum_{i=1}^{k} n_i = n$$

A component with n_i vertices will have the maximum possible number of edges when it is complete.

i.e., it contains $\frac{1}{2} n_i (n_i - 1)$ edges.

The maximum number of edges is

$$\begin{aligned}
\frac{1}{2}\sum_{i=1}^{k} n_i(n_i-1) &= \frac{1}{2}\sum_{i=1}^{k} n_i^2 - \frac{1}{2}\sum_{i=1}^{k} n_i \\
&\le \frac{1}{2}\left[n^2 - (k-1)(2n-k)\right] - \frac{1}{2}n \\
&= \frac{1}{2}\left[n^2 - 2nk + k^2 + n - k\right] \\
&= \frac{1}{2}(n-k)(n-k+1)
\end{aligned}$$

Suppose graph is disconnected. Then it has at least two components,

Therefore, $$m \le \frac{1}{2}(n-k)(n-k+1) \text{ for } k \ge 2.$$

$$\le \frac{1}{2}(n-2)(n-1)$$

This contradiction assumes that

$$m = \frac{1}{2}(n-1)(n-2)$$ ■

Therefore, the graph is connected.

Theorem 7: Every graph $G = (V, E)$ has at least $|V| - |E|$ connected components.

Proof: We use induction on the number of edges. Let $P(n)$ be the proposition that every graph $G = (V, E)$ with $|E| = n$ has at least $|V| - n$ connected components.

Base case: In a graph with 0 edges, each vertex is itself a connected component, and so there are exactly $|V| - 0 = |V|$ connected components.

Inductive step: Now we assume that the induction hypothesis holds for every n-edge graph in order to prove that it holds for every $(n + 1)$ — edge graph, where $n \geq 0$. Consider a graph G (V, E) with $n + 1$ edges. Remove an arbitrary edge u-v and call the resulting graph G'. By the induction assumption, G' has at least $|V| - n$ connected components. Now add back the edge $u - v$ to obtain the original graph G. If u and v were in the same connected component of G', then G has the same number of connected components as G', which is at least $|V| - n$. Otherwise, if u and v were in different connected components of G', which is at least $|V| - n$. Otherwise, if u and v were in different connected components of G', then these two components are merged into one in G, but all other components remain. Therefore, G has at least $|V| - n - 1 = |V| - (n + 1)$ connected components. ■

Corollary: *Every connected graph with n vertices has at least n – 1 edges.*

A couple points about the proof of Theorem 7 are worth noting. First, notice that we used induction on the number of edges in the graph. This is very common in proofs involving graphs, and so is induction n the number of vertices.

The second point is more subtle. Notice that in the inductive step, we took an arbitrary $(n + 1)$-edge graph, threw out an edge so that we could apply the induction assumption, and then put the edge back. This might seem like needless effort; why not start with an n-edge graph and add one more to get an $(n + 1)$-edge graph? That would work fine in this case, but opens the door to a very nasty logical error in similar arguments. ■

Theorem 8: A connected graph has an Euler tour if and only if every vertex has even degree.

Proof: If a graph has an Euler tour, then every vertex must have even degree; in particular, a vertex visited k times on an Euler tour must have degree $2k$.

Now suppose every vertex in graph G has even degree. Let W be the longest walk in G that traverses every edge at most once:

$$W = v_0, v_0 - v_1, v_1, v_1 - v_2, v_2, \ldots, v_{n-1} - v_n, v_n$$

The walk W must traverse every edge incident to v_n; otherwise, the walk could be extended. In particular, the walk traverses two of these edges each time it passes through v_n and traverses $v_{n-1} - v_n$ at the end of the walk. This accounts for an odd number of edges, but the degree of v_n is even by assumption. Therefore, the walk must also begin at v_n; that is, $v_0 = v_n$.

Suppose that W is not an Euler tour. Because G is a connected graph, we can find an edge not in W but incident to some vertex in W. Call this edge $u - v_i$. But then we can construct a longer walk:

$$u, u - v_i, v_i, v_i - v_i - v_{i+1}, \ldots, v_{n-1} - v_n, v_n, v_0 - v_1, \ldots, v_{i-1} - v_i, v_i$$

This contradicts the definition of W, so W must be an Euler tour after all.

Corollary: *A connected graph has an Euler walk if and only if either 0 or 2 vertices have odd degree.*

Hamiltonian cycles are the unruly cousins of Euler tours. A Hamiltonian cycle is walk that starts and ends at the same vertex and visits every vertex in a graph exactly once. There is no simple characterization of all graphs with a Hamiltonian cycle. (In fact, determining whether a given graph has a Hamiltonian cycle is "NP-complete".) ■

Theorem 9: Let G be a digraph (possibly with self-loops) with vertices $v_1,\ldots,v_n$. Let M be the adjacency matrix of G. Then M_{ij}^k is equal to the number of length-k walks from v_i to v_j.

Proof: We use induction on k. The induction hypothesis is that M_{ij}^k is equal to the number of length-k walks from v_i to v_j, for all i, j.

Each vertex has a length-0 walk only to itself. Since $M_{ij}^k = 1$ if and only if $i = j$, the hypothesis holds for $k = 0$.

Now suppose that the hypothesis holds for some $k \geq 0$. We prove that it also holds for $k + 1$. Every length-$(k + 1)$ walk from v_i to v_j consists of a length k walk from v_i to some intermediate vertex v_m followed by an edge v_m–v_j. Thus, the number of length-$(k + 1)$ walks from v_i to v_j is equal to:

$$M_{iv1}^k M_{v1j} + M_{iv2}^k M_{v2j} + \ldots + M_{ivn}^k M_{vnj}$$

This is precisely the value of M_{ij}^{k+1}, so the hypothesis holds for $k + 1$ as well. The theorem follows by induction. ■

10.14 Some Popular Problems in Graph Theory

10.14.1 Tournament Ranking Problem

Suppose that n players compete in a round-robin tournament. Thus, for every pair of players u and v, either u beats v or else v beats u. Interpreting the results of a round-robin tournament can be problematic. There might be all sorts of cycles where x beat y, y beat z, yet z beat x. Graph theory provides at least a partial solution to this problem.

The results of a round-robin tournament can be presented with a tournament graph. This is a directed graph in which the vertices represent players and the edge indicate the outcomes of games. In particular, an edge from u to v indicates that player u defeated player v. In a round-robin tournament, every pair of players has a match. Thus, in a tournament graph there is either can edge from u to v or an edge from v to u for every pair of vertices u and v. Here is an example of a tournament graph (refer Fig. 10.27)

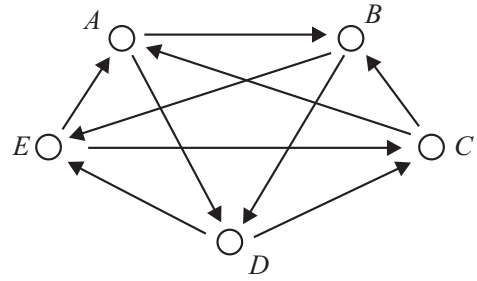


Fig. 10.27: Tournament graph

The notions of walks, Euler tours, and Hamiltonian cycles all carry over naturally to directed graphs. A directed walk is an alternating sequence of vertices and directed edges:

$v_0, v_0 \rightarrow v_1, v_1, v_1 \rightarrow v_2, v_2, ..., v_{n-1}, v_{n-1} \rightarrow v_n, v_n$

A directed Hamiltonian path is a directed walk that visits every vertex exactly once.

We're going to prove that in every round-robin tournament, there exists a ranking of the players such that each player lost to the player ranked one position higher. For example, in the tournament above, the ranking.

$$A > B > D > E > C$$

satisfies this criterion, because B lost to A, D lost to B, E lost to D and C lost to E. In graph terms, proving the existence of such a ranking amounts to proving that every tournament graph has a Hamiltonian path.

Theorem 10: Every tournament graph contains a directed Hamiltonian path.

Proof: We use strong induction. Let $P(n)$ be the proposition that every tournament graph with n vertices contains a directed Hamiltonian path.

Base case: $P(1)$ is trivially true; every graph with a single vertex has a Hamiltonian path consisting of only that vertex.

Inductive step: For $n \geq 1$, we assume that $P(1),...,P(n)$ are all true and prove $P(n + 1)$. Consider a tournament with $n + 1$ players. Select one vertex v arbitrarily. Every other vertex in the tournament either has an edge to vertex v or an edge from vertex v. Thus, we can partition the remaining vertices into two corresponding sets, T and F, each containing at most n vertices. (Fig. 10.28)

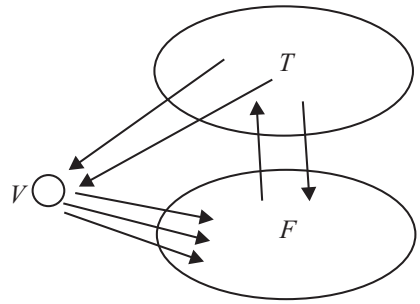


Fig. 10.28: Tournament Graph as Hamiltonian Path

The vertices in T together with the edges that join them form a smaller tournament. Thus, by strong induction, there is a Hamiltonian path within T. Similarly, there is a Hamiltonian path within the tournament on the vertices in F. Joining the path in T to the vertex v followed by the path in F gives a Hamiltonian path through the whole tournament. (As special cases, if T or F is empty, then so is the corresponding portion of the path.)

Note: The ranking defined by a Hamiltonian path is not entirely satisfactory. In the example tournament, notice that the lowest-ranked player (C) actually defeated the highest-ranked player (A).

10.14.2 The Königsberg Bridge Problem

The Königsberg bridge problem is perhaps the best-known example in Graph Theory. It was a long-standing problem until solved by Leonhard Euler (1707-1783) in 1736, by means of a graph. Euler wrote the first paper even in graph theory and thus became the origination of Graph Theory as well as of the rest of topology. The city of Königsberg was located on the Pregel river in Prussia. The city occupied the island of kneiphopf plus areas on both banks. These regions were linked by seven bridges as shown in Fig. 10.29. The citizens wondered whether they could leave home, cross every bridge exactly once, and return home. The problem reduces to traversing the Fig. (right) with heavy dots representing land masses and cures representing bridges.

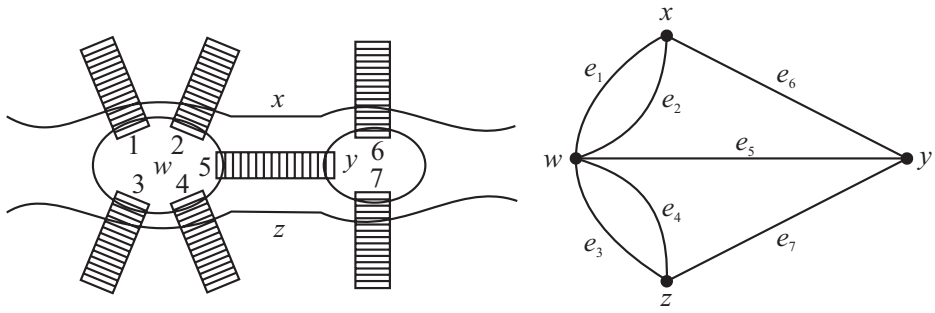


Fig. 10.29: The Königsberg Bridge Problem

Two stands *w* and *y*, formed by the Pregel river in Königsberg (Then the capital of east Prussia but now renamed Kalinongrad and in West Soviet Russia) were connected to each other to the banks *x* and *z* with seven budges as shown in Fig. The problem was to start any of the four land areas of the city *w*, *x*, *y* or *z*, walk over each of the seven bridges exactly once, and return to the starting point (without swimming across the river). Euler represented this situation by means of a graph as shown in Fig.. The vertices represent the land areas and the edges represent the bridges.

Later on the problem was solved by introducing a new bridge on left most side.

10.14.3 Four Colour Problem

One of the most famous problem of Graph Theory is the four colour problem. This problem states that any map on a plane on the surface of

a sphere can be coloured with four colours in such a way that no two adjacent countries or states have the same colour. This problem can be translated as a problem in Graph Theory. We represent each country or state by a point and join two points by a line if the countries are adjacent. The problem is to colour the points in each way that adjacent points have different colours. This problem was first posed in 1852 by Frances Gutherie a post-graduate student at the University College, London. This problem was finally proved by Appel and Hahen in 1976 and they have used 400 pages of arguments and about 1200 hours of computer time on some of the best computers in the world to arrive the solution.

10.14.4 Three Utilities Problem

We consider three houses H_1, H_2 and H_3, each to be connected to each of the three utilities viz. water (W), gas (G), and electricity (E) by means of conducts. It is possible to make such connections without any crossovers of the conduits. We can see in Fig. 10.30. How this problem can represented by a graph the conduits are shown as edges while the houses and utilities supply centres are vertices. The graph (right) cannot be drawn n the plane without edge crossing over. The answer of this problem is no.

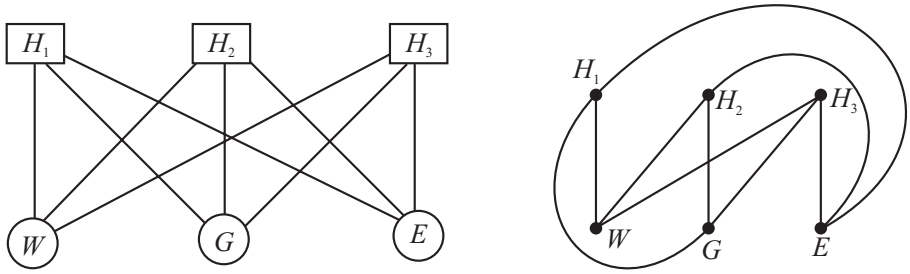


Fig. 10.30: Three Utilities Problem

10.14.5 Traveling - Salesman Problem

A salesman requires to visit a number of cities during his assignment. Given the distance between the cities, in what order should be travel as to visit every city precisely once and return home, with the minimum mileage travelled?

We represent the cities by vertices and the road between them by edges. We get a graph.

In this graph, with every edge e_i there is associated a real number $W(e_i)$ being the weight of edge e_i.

Illustration:

Suppose that a salesman wants to visit five cities, namely A, B, C, D and E (as shown in Fig. 10.31). In which order should he visit these cities to travel the minimum total distance? To solve this problem we may assume the salesman starts in A (since this must be a part of circuit) and examine all possible ways for him to visit the other four cities and then return to A. There exist a total of 24 such circuits, but since we travel the same distance when we travel a circuit in reverse order, we need to consider only 12 different circuits to find the minimum total distance he must travel. We can list these 12 different circuits and the total distance travelled for each circuit. The routes can be found the distance travelled. The travelling salesman problem asks for a circuit of minimum total weight in a weighted, complete, undirected graph that visits each vertex exactly once and returns to its starting point. This is equivalent to asking for a Hamiltonian circuit with minimum total weight in the complete graph, since each vertex is visited exactly once in the circuit.

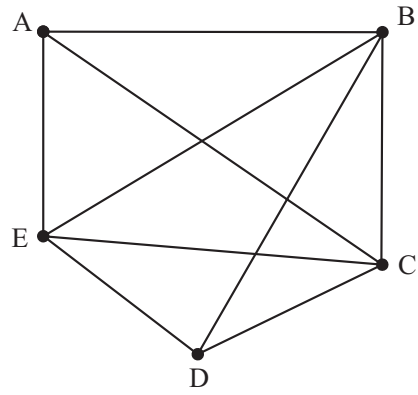


Fig. 10.31: Traveling Salesman Problem

The most straight way to solve an instance of this problem is to examine all possible Hamiltonion circuits and select one of minimum total length. (Fig. 10.31)

If we have n vertices in the graph. Once a starting point is chosen, there are $(n–1)!$ different Hamiltonion circuits to be examined, since there are $(n–1)$ choices for second vertex, $(n – 2)$ choices for the third vertex and, so on.

A Hamiltonion circuit can be travelled in reverse order, we can only examine $\frac{(n-1)!}{2}$ circuits to reach the answer. $\frac{(n-1)!}{2}$ grown extremely rapidly.

10.14.6 MTNL'S Networking Problem

Suppose Mahanagar Telephone Nigam Limited (MTNL) is interested in identifying those lines that must stay in services to avoid disconnecting the network as shown in Fig. 10.32. Which is a typical problem of graph theory.

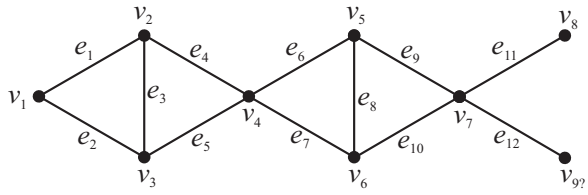


Fig. 10.32: MTNL Network Problem

10.14.7 Electrical Network Problems

The properties of an electrical network are functions of only two key factors:

(*i*) The nature and value of the elements forming the network, such as resistors, inductors and transistors etc.

(*ii*) The way these elements are connected together *i.e.* the topology of the network.

Since these are a few different types of elements of electrical circuit, the variations in networks are due to variations in networks are due to variation in technology.

The particular topology of an electrical network is studied by means of graph theory. While drawing a graph of an electrical network the junctions are represented by vertices, and the branches are represented by edges, regardless the nature and size of the electrical elements shown in Fig. 10.33.

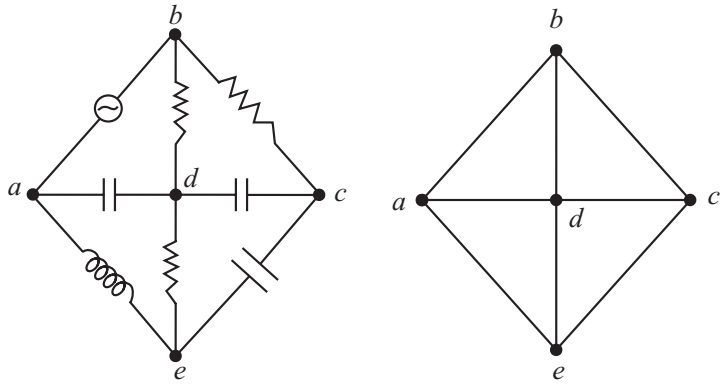


Fig. 10.33: Electrical Network Problem

10.14.8 Satellite Channel Problem

Suppose six TV companies apply for frequency allotment. If the relay centre of these companies are not less than 1000 km then only the same frequency could be alloted without interference. The aim is to assign as small number of different frequency as possible. The Fig. 10.34 shows the line joining two companies to show that they came relay centres less than 1000 km apart. This is a typical problem of graph theory. As an obvious case, we could assign same frequency to company 1 and 4, 2 and 5, and 3 and 6 respectively.

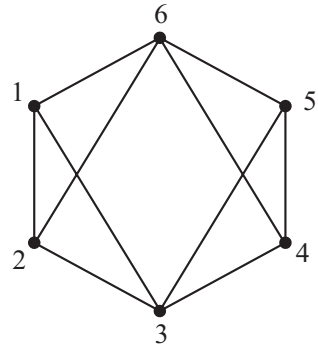


Fig. 10.34: Relay Centres and their Distance

10.15 Applications of Graphs

Graphs are the most useful mathematical objects in computer science. We can model an enormous number of real-world systems and phenomena using graphs. Once we have created such a model, we can tap the vast

store of theorems about graphs to gain insight into the system we are modeling. Here are some practical situations where graphs arise:

Data Structures: Each vertex represents a data object. There is a directed edge from one object to another if the first contains a pointer or reference to the second.

Attraction: Each vertex represents a person, and each edge represents a romantic attraction. The graph could be directed to model the unfortunate asymmetries.

Airline Connections: Each vertex represents an airport. If there is a direct flight between two airports, then there is an edge between the corresponding vertices. These graphs often appear in airline magazines.

The Web: Each vertex represents a web page. Directed edges between vertices represent hyperlinks.

People often put numbers on the edges of graph, put colors on the vertices, or add other ornaments that capture additional aspects of the phenomenon being modeled. For example, a graph of airline connections might have numbers on the edges to indicate the duration of the corresponding flight. The vertices in the attraction graph might be colored to indicate the person's gender.

SOLVED EXAMPLES

Example 10

Determine the number of edges in a graph with 6 vertices, 2 of degree 4 and 4 of degree 2. Plot two such graphs.

Solution

We consider a graph with 6 vertices and having e number of edges from Handshaking lemm, we have.

$\Rightarrow \quad d(v_1) + d(v_2) + d(v_3) + d(v_4) + d(v_5) + d(v_6) = 2e.$

given, 2 vertices are of degree 4 and 4 vertices are of degree 2.

Hence the above equation

$$(4 + 4) + (2 + 2 + 2 + 2) = 2e.$$

$\Rightarrow \qquad 16 = 2e \quad \Rightarrow \quad e = 8.$

Hence the number of edges in a graph with 6 vertices with the given conditions is 8.

E.g.

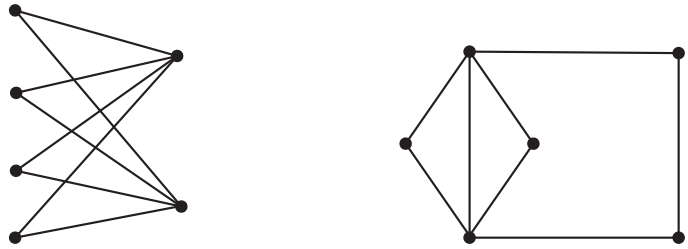
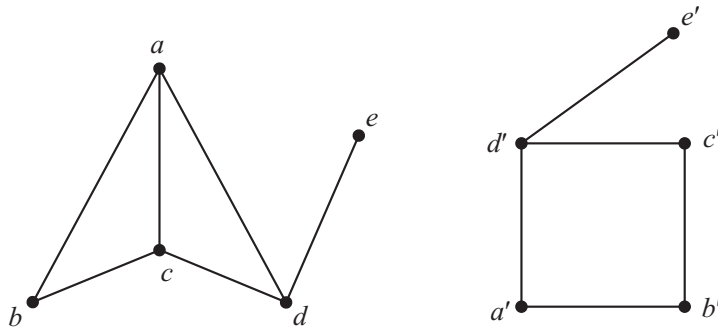


Fig. 10.35

■

Example 11

Show that the following graphs are isomorphic [UPTU 2006]



(G) (G')

Fig. 10.36

Solution

We define the function $f: G \to G'$ between two graphs.

The following are the degree of G and G'

deg. (G)	deg (G')
$\deg(a) = 3$	$\deg(a') = 3$
$\deg(b) = 2$	$\deg(b') = 2$
$\deg(c) = 3$	$\deg(c') = 3$
$\deg(d) = 3$	$\deg(d') = 3$
$\deg(e) = 1$	$\deg(e') = 1$

Each graph has 5-vertices and 6-edges,

$$d(a) = d(a') = 3$$
$$d(b) = d(b') = 2$$
$$d(c) = d(c') = 3$$
$$d(d) = d(d') = 3$$
$$d(e) = d(e') = 1$$

Hence the correspondence is $a - a'$, $b - b'$,...,$e - e'$
which is one-to-one

Hence the given graphs are isomorphic. ■

Example 12

Find the rank and nullity of the complete graph k_n.

Solution

Since k_n is a connected graph with n vertices

$$m = \frac{n(n-1)}{2} \text{ edges.}$$

Therefore, by the definition of rank and nullity, we have

Rank of $k_n = n - 1$

Nullity of $k_n = m - n + 1 = \frac{1}{2}n(n-1) - n + 1$

$$= \frac{1}{2}n(n-1) - n + 1 \quad \textbf{Ans.}$$ ■

Example 13

If G be a simple graph with n vertices and m edges where m is at least 3.

If $m \geq \frac{1}{2}$ *(n – 1)(n – 2) + 2.*

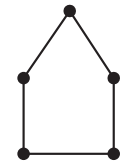
Prove that G is Hamiltonian. Is the converse true?

Solution

Let u and v be any two non-adjacent vertices in G, and x, y be their respective degrees.

If we delate u, v from G, we get a subgraph with $(n-2)$ vertices.

If this subgraph has q edges, then $q \le \frac{1}{2}(n-2)(n-3)$



Since u and v are non-adjacent, then

$$m = q + x + y.$$

Thus, $x + y = m - q \ge \left\{\frac{1}{2}(n-1)(n-2)+2\right\} - \left\{\frac{1}{2}(n-2)(n-3)\right\}$

$= n$

Hence the graph is Hamiltonion.

The converse of the result may or may not true because, a 2-regular graph with 5-vertices is Hamiltonian but the inequality does not hold.

■

SUMMARY

1. A **graph** G consists of a nonempty, finite set V of vertices and a set E of edges connecting them: $G = (V, E)$. An edge connecting vertices u and v is denoted by (u, v).
2. A graph with no loops or parallel edge is a **simple graph**.
3. Two vertices v and w in a graph G are **adjacent** if (v, w) is an edge in G.
4. The **degree** of a vertex v is the number of edges meeting at v.
5. The **adjacency matrix** of G is an $n \times n$ matrix $A = (a_{ij})$, where a_{ij} = the number of edges from vertex v_i to vertex v_j.
6. Let e denote the number of edges of a graph with n vertices v_1, v_2, ..., v_n. Then $\sum_{i=1}^{n} \deg(v_i) = 2e$
7. A **Subgraph** of a graph $G = (V, E)$ is a graph $H = (V_1, E_1)$, where $V_1 \subseteq V$ and $E_1 \subseteq E$.
8. A simple graph with n vertices is **complex graph** K_n if every pair of distinct vertices is connected by an edge.
9. Let $G = (V, E)$ be a simple graph such that $V = V_1 \cup V_2$, $V_1 \cup V_2 = \varnothing$, and every edge is incident with a vertex in V_1 and V_2. G is a **bipartite graph**. If every vertex in V_1 is adjacent to every vertex in

V_2, G is a **complete bipartite graph**. If $|V_1| = m$ and $|V_2| = n$, then G is denoted by $K_{m,n}$.

10. A **weighted graph** is a simple graph in which every edge is assigned a positive number, called the **weight** of the edge
11. An ***r*-regular graph** is a simple graph in which every vertex has the same degree r
12. The **complement** $G' = (V, E')$ of a simple graph $G = (V, E)$ contains all vertices in G. An edge $(u,v) \in E'$ if and only if $\{u, v\} \notin E$.
13. Two simple graphs, $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, are **isomorphic** if a bijection $f : V_1 \rightarrow V_2$, exists such that $\{u, v\} \in E_1$ if and only if $\{f(u), f(v)\} \in E_2$. The function f is an **isomorphism** between G_1 and G_2. If G_1 and G_2 are isomorphic, they have exactly the same properties.
14. An **isomorphism invariant** is a property shared by isomorphic graphs.
15. A **path of length** n from a vertex v_0 to a vertex v_n is a sequence of vertices v_i and edges e_i of the form $v_0 - e_1 - v_1 \ldots e_n - v_n$, where each edge e_i is incident with the vertices $v_i - 1$ and v_i $(1 \leq i \leq n)$. A **simple** path contains no repeated vertices, except perhaps the endpoints.
16. A path from v_0 to v_n is **closed** if $v_0 = v_n$. Otherwise, it is **open.**
17. A **cycle** is a simple closed path.
18. A **circuit** is a simple closed path with no repeated edges.
19. A **connected graph** contains a path between any two distinct vertices.
20. The length of a simple path between any two vertices of a connected graph is at most $n - 1$.
21. If A is the adjacency matrix of a connected graph, the number of paths of length k from vertex v_i to v_j is given by ijth entry of A^k, where $1 \leq k \leq n - 1$.
22. A simple path in connected graph is **Hamiltonian** if it contains every vertex.
23. A cycle in a connected graph is **Hamiltonian** if it contains every vertex.
24. A connected graph that contains a Hamiltonian cycle is **Hamiltonian.**

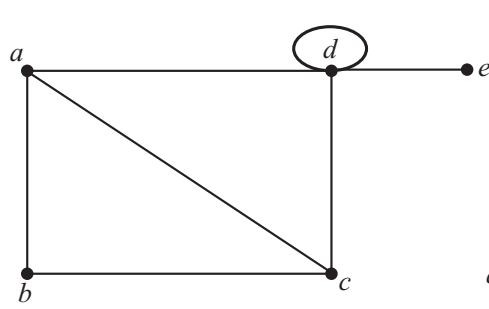
EXERCISE

1. How many vertices are needed to construct a graph with 6 edges in which each vertex is of degree? **[Ans.** 6**]**

2. Show that the degree of a vertex of a simple graph G on n-vertices cannot exceed $n - 1$

3. What is the size of an r-regular (p, q) graph? **[Ans.** $q = \frac{p \times r}{2}$ **]**

4. Show that the following graphs are isomorphic.



(G) $(G)'$

5. Let $Q = (V, E, \phi)$ be the graph where

$V = \{A, B, C, D, E, F, G, H\}$, $E = \{a, b, c, d, e, f, g, h, i, j, k, l\}$

and $$\phi = \begin{pmatrix} a & b & c & d & e & f & g & h & i & j & k & l \\ A & A & D & E & A & E & B & F & G & C & A & E \\ B & D & E & B & B & G & F & G & C & C & A & G \end{pmatrix}$$

What is the degree sequence of Q?

6. Consider the following unlabeled pictorial representation of Q

(a) Create a pictorial representation of Q by labeling $P'(Q)$ with the edges and vertices of Q.

(b) A necessary condition that a pictorial representation of a graph R can be created by labeling $P'(Q)$ with the vertices and edges of R is that the degree sequence of R be (0, 2, 2, 3, 4, 4, 4, 5).

True or false? Explain

(*c*) A sufficient condition that a pictorial representation of a graph *R* can be created by labeling $P'(Q)$ with the vertices and edges of *R* is that the degree sequence of *R* be (0, 2, 2, 3, 4, 4, 5). True or false? Explain.

7. In each of the following problems information about the degree sequence of a graph is given. In each case, decide if a graph satisfying the specified conditions exists on not. Give reason in each case.

(*a*) A graph *Q* with degree sequence (1, 1, 2, 3, 3, 5)?

(*b*) A graph *Q* with degree sequence (1, 2, 2, 3, 3, 5), loops and parallel edges allowed?

(*c*) A graph *Q* with degree sequence (1, 2, 2, 3, 3, 5), no but parallel edges allowed?

(*d*) A graph *Q* with degree sequence (1, 2, 2, 3, 3, 5), no loops or parallel edges allowed?

(*e*) A simple graph *Q* with degree sequence (3, 3, 3, 3)?

(*f*) A graph *Q* with degree sequence (3, 3, 3, 3), no loops or parallel edges allowed?

(*g*) A graph *Q* with degree sequence (4, 4, 4, 4, 4), no loops or parallel edges allowed?

(*h*) A graph *Q* with degree sequence (4, 4, 4, 4, 4), no loops or parallel edges allowed?

(*i*) A graph *Q* with degree sequence (4, 4, 4, 4, 6), no loops or parallel edges allowed?

8. Divide the following graphs into isomorphism equivalence classes and justify your answer; *i.e.*, explain why you have the classes that you do. In all cases $V = 4$.

(*a*) $\phi = \begin{pmatrix} a & b & c & d & e & f \\ \{1,2\} & \{1,2\} & \{2,3\} & \{3,4\} & \{1,4\} & \{2,4\} \end{pmatrix}$

(*b*) $\phi = \begin{pmatrix} A & B & C & D & E & F \\ \{1,2\} & \{1,4\} & \{1,4\} & \{1,2\} & \{2,3\} & \{3,4\} \end{pmatrix}$

(*c*) $\phi = \begin{pmatrix} u & v & w & x & y & z \\ \{2,3\} & \{1,3\} & \{3,4\} & \{1,4\} & \{1,2\} & \{1,2\} \end{pmatrix}$

$$(d)\ \phi = \begin{pmatrix} P & Q & R & S & T & U \\ \{3,4\} & \{2,4\} & \{1,3\} & \{3,4\} & \{1,2\} & \{1,2\} \end{pmatrix}$$

9. A graph $G = (V, E)$ is called bipartite if V can be partitioned into two sets C and S such that each edge has one vertex in C and one vertex in S. As a specific example, let C be the set of courses at the university and S the set of students. Let $V = C \cup S$ and let $\{s, c\} \in E$ if and only if student s is enrolled in course c.

(*a*) Prove that $G = (V, E)$ is a simple graph.

(*b*) Prove that every cycle of G has an even number of edges.

10. An *oriented simple graph* is a simple graph which has been converted to a digraph by assigning an orientation to each edge. The orientation of $\{u, v\}$ can be thought of as a mapping of it to either (u, v) or (v, u).

(*a*) Give an example of a simple digraph that has no loops but is not an oriented simple graph.

(*b*) Find the number of oriented simple digraphs.

(*c*) Find the number of them with exactly q edges.

11. A binary relation R on S is an order relation if it is reflexive, antisymmetric, and transitive. R is antisymmetric if for $(x, y) \notin R$ with $x \neq y$, $(y, x) \in R$. Given an order relation R, the covering relation H of R consists of all $(x, z) \in R$, $x \neq z$, such that there is no y, distinct from both x and z, such that $(x, y) \in R$ and $(y, z) \in R$. A pictorial representation of the covering relation as a directed graph is called a "Hasse diagram" of H.

(*a*) Show that the divides relation on

$$S = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16\}$$

Is an order relation. By definition, (x, y) is in the divides relation on S is x is a factor of y. Thus, (4, 12) is in the divides relation. $x| y$ is the standard notation for x is a factor of y.

(*b*) Find and draw a picture of the directed graph of the covering relation of the divides relation.

Hint: You must find all pairs $(x, z) \in S \times S$ such that $x| y$ but there does not exist any 4, $x < y < z$, such that $x| y$ and $y|z$.

Suggested Readings

1. N. Cristofides, *Graph Theory*: *An Algorithmic Approach*, Academic Press, New York, 1975.
2. B.W. Jackson and D.Thoro, *Applied Combinatorics with Problem Solving,* Addison-Wesley, Reading, MA, 1990, pp. 134-200.
3. J.A. McHugh, *Algorithmic Graph Theory*, Prentice-Hall, Englewood Cliffs, NJ, 1990.
4. A. Ralston, "Debrujin Sequences—A model Example of the Interaction of Discrete Mathematics and Computer Science," *Mathematics Magazines,* Vol. 55 (May 1982), pp. 131-143.
5. S.S.Skieno, *Implementing Discrete Mathematics,* Addison-Wesley, Reading, MA, 1990.
6. K.Thulasiraman and M.N.S. Swamy, *Graphs*: *Theory and Algorithms,* Wiley, New York, 1992.

❑❑❑

11

Trees

Gustav Robert Kirchoff (1824–1887)

Gustav Robert Kirchoff (1824-1887) was a German physicist born in Konigsberg, Prussia. His father was a lawyer. He graduated in early age of 18 from the local gymnasium from the university of Konigsberg and received his doctorate five years later. He started teaching in Berlin in 1848 and two years later he joined as a faculty at the university of Breslan where he met well known Chemist Robert Bunsen. In 1854 both moved to Heidelberg where he made the greatest contribution to science (Kirchoff's laws)

In 1875, Kirchoff accepted the chair of theoretical physics at the university of Berlin which he hold until his death.

Kirchoff made significant contributions to every branch of physics and engineering. Some of his results are widely used in data structure and graph theory.

11.1 Introduction

The word "tree" suggest branching out from a root and never completing a cycle. As a graph, trees have many applications especially in data storage, searching, and communication. Trees play an important role in a variety of algorithms. We use decision trees to enhance our understanding of recursion. We can point out some of its applications to simple situations and puzzles and games, deferring the applications to more complex scientific problems.

Kirchhoff developed the theory of trees in 1847, in order to solve the system of simultaneous linear equations which give the current in each branch and around each circuit of an electric network. In 1857, *Cayley*

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_11

discovered the important class of graphs called trees by considering the changes of variables in the differential calculates. Later on, he was engaged in enumerating the isomers of saturated hydro-carbons (C_nH_{2n+2}).

An acyclic graph, one not containing any cycle, is called a forest. A connected forest is called a tree. Thus, a forest is a graph whose components are trees. The vertices of degree one in a tree are its leaves. Every non trivial tree has atleast two leaves-take, for example, the ends of a longest path. This little fact often comes in handy, especially in induction proofs about trees: If we remove a leaf from a tree, what remains is still a tree (see Fig. 11.1)

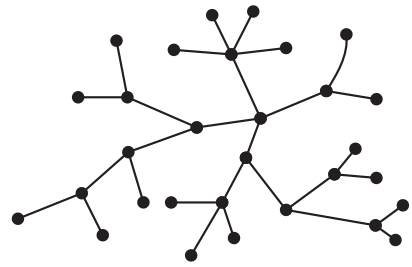


Fig. 11.1: A Tree

11.2 Definitions of a Tree

If G is a connected graph without any cycles then G is called a tree. A tree is also called a free tree. If $|V| = 1$, then G is connected and hence it is a tree.

Theorem 1: Alternative definitions of a tree.

If G is a connected graph, the following are equivalent:

(i) G is a tree.

(ii) G has no cycles.

(iii) For every pair of vertices $u \neq v$, there is exactly one path from u to v.

(iv) Removing any edge from G gives a graph which is not connected.

(v) The number vertices of G is one more than the number of edges of G.

Proof: We are given that, G is connected, thus by definition of a tree (i) and (ii) are equivalent we have that two vertices $u \neq v$ are on a cycle of G iff there are atleast two paths from u to v that have no vertices in common except the end points u and v. from the above statement (ii) $\Rightarrow$ (iii)

If $\{u, v\}$ is an edge, it follows from (iii) that the edge is only path from u to v and removing it disconnects the graph have (iii) $\Rightarrow$ (iv).

We have seen that (*i*) and (*ii*) are equivalent, and we also have seen that (*ii*) $\Rightarrow$ (*iii*) $\Rightarrow$ (*iv*) $\Rightarrow$ (*v*), so (*i*), (*ii*), (*iii*) and (*iv*) are equivalent. All that remains is to include (*v*) in this equivalence class of statements. ■

11.3 Forest

A forest is a graph all of whose connected components are trees. In particular, *a forest with single component is a tree.*

Example 1

A Relation for Forests

Suppose a forest has v vertices, e edges and c (connected) components. What values are possible for the triple of numbers (v, e, c)? It might seem at first that almost anything is possible, but this is not so. In fact $v - c = e$. Why? Let the forest consist of trees T_1, ..., T_c and let the triples for T_i be (v_i, e_i, c_i). Since a tree is connected, $C_i = 1$. By the theorem, $e_i = v_i - 1$. Since $v = v_1 + ...+ v_c$ and $e = e_1 + ,..., + e_c$ we have

$$\begin{aligned} e &= (v_1 - 1) + (v_2 - 1) + ... + (v_c - 1) \\ &= (v_1 + ...+ v_c) - c = v - c. \end{aligned}$$

Suppose a forest has $e = 12$ and $v = 15$. We know immediately that it must be made up of three trees because $c = v - e = 15 - 12$.

Suppose we know that a graph $G = (V, E, \phi)$ has $v = 15$ and $c = 3$, what is the fewest edges it could have? For each component of G, we can remove edges one by one until we cannot remove any more without breaking the component into two components. At this point, we are left with each component a tree. Thus we are left with a forest of $c = 3$ trees that still has $v = 15$ vertices. By our relation $v - c = e$, this forest has 12 edges. Since we may have removed edges from the original graph to get to this forest, the original graph has at least 12 edges.

What is the maximum number of edges that a graph $G = (V, E, \phi)$ with $v = 15$ and $c = 3$ could have? Since we allow multiple edges, a graph could have an arbitrarily large number of edges for a fixed v and c. if e is an edge with $\phi(e) = \{u, v\}$, add in as many edges e_i with $\phi(e_i) = \{u, v\}$ as we wish. Hence we will have to insist that G be a simple graph.

What is the maximum number of edges that a simple graph G with $v = 15$ and $c = 3$ could have? We start with a graph where c is not specified.

The edges in a simple graph are a subset of $P_2(V)$ and since $P_2(V)$ has $\binom{V}{2}$ elements, a simple graph with v vertices has at most $\binom{V}{2}$ edges.

We return to the case when we know there must be three components in our simple graph. Suppose the number of vertices in the components are v_1, v_2, and v_3. Since there are no edges between components, we can look at each component by itself. Using the result in the previous paragraph for each component, the maximum number of possible edges is $\binom{v_1}{2}+\binom{v_2}{2}+\binom{v_3}{2}$. We don't know v_1, v_2, v_3. All we know is that they are strictly positive integers that sum to v. It turns out that the maximum occurs when one of v_i is as large as possible and the others equal 1, but the proof is beyond this course. Thus the answer is $\binom{v-2}{2}$, which in our case is $\binom{13}{2} = 78$. In general, if there were c components, $c - 1$ components would have one vertex each and the remaining component would have $v - (c - 1) = v + 1 - c$ vertices. Hence there can be no more than $\binom{v+1-c}{2}$ edges.

The above results can conclude that:

(*i*) There is no graph $G = (V, E, \phi)$ with $v - c > e$

(*ii*) If $v - c = e$, the graph is a forest of c trees and any such forest will do as an example.

(*iii*) If $v - c < e$, there are many examples, none of which are forests.

(*iv*) If $v - c < e$ and we have a simple graph, then we must have $e \leq \binom{v+1-c}{2}$. ■

11.4 Rooted Graph

A pair (G, v), consisting of a graph $G = (V, E, \phi)$ and a specified vertex v, is called a rooted graph with root v.

A rooted tree T with the vertex set V is the tree that can be defined recursively as:

(*i*) T has a specially designated vertex $v_1 \in V$, called the root of T. The subgraph of T_1 consisting of the vertices $V - \{v\}$ is partitionade into subgraphs.

(*ii*) $T_1, T_2, ...T_r$ each of which is itself a rooted tree. Each one of these r-rooted tree is called a subtree of v_1 (as shown in Fig. 11.2)

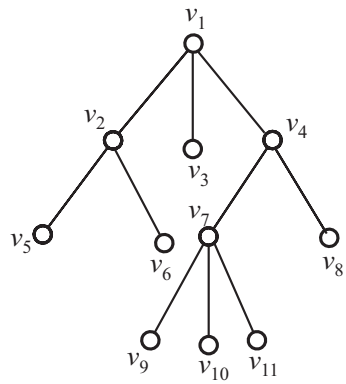


Fig. 11.2: A Rooted Tree

Note: A tree T^* of a spanning tree T in a connected graph G is the spanning subgraph of G containing exactly those edges of G which are not in T. The edges of G which are not in T^* are called its twigs.

11.5 Parent, Child, Sibling and Leaf

Let (T, r) be a rooted tree. If w is any vertex other than r, let $r = v_0$, $v_1, v_2, ..., v_k, v_{k+1} = w$, be the list of vertices on the unique path from r to w. We can say v_k is parent of w and w a child of v_k parents and children are also called father and sons. Vertices with the same parent are siblings. A vertex with no children is called a leaf. All the other vertices are internal vertices of the tree.

11.6 Rooted Plane Tree

Let (T, r) be a rooted tree. For each vertex, order the children of the vertex. The result is a rooted plane tree, which is abbreviated as *RP*-tree. *RP*-trees are also called ordered trees. An *RP*-tree is also called, a decision tree, and, when there is no chance of misunderstanding, simply a tree.

Example 2

A Rooted Plane Tree

Fig. 11.3 is a picture of a rooted plane tree $T = (V, E, \phi)$. In this situation $V = 11$ and $E = \{a, b, c, d, e, f, g, h, i, j\}$. There are no parallel

edges or loops. The root is $r = 1$. For each vertex, there is a unique path from the root to that vertex. Since ϕ is an injection, once ϕ has defined that unique path can be specified by the vertex sequence alone. The path from the root is to 6 is (1, 3, 6). The path from the root to 9 is (1, 3, 6, 9) (In computer science we refer to the path from the root to vertex v as the "stack" if v.

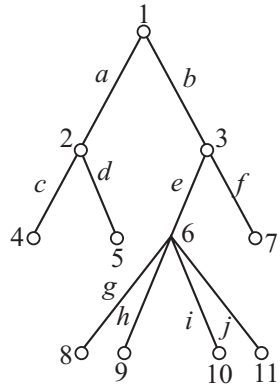


Fig. 11.3: RP – Tree

The vertex 6 is the parent of the vertex 9. The vertices 8, 9, 10 and 11 are the children of 6 and, they are siblings of each other. The leaves of the tree are 4, 5, 7, 8, 9, 10, 11. All other vertices (including root) are the internal vertices of the tree.

We must remember that, an *RP*-tree is a tree with added properties. So, we must satisfy, T has no cycle. There is a unique path between any two vertices. Removing any edge gives a graph which is not connected. (e. g. removing j disconnects T into a tree with 10 vertices and a tree with 1 vertices; removing e disconnects T into a tree with 6 vertices and one with 5 vertices). At last, the number of edges (10) is one less than the number of vertices. ■

Example 3

Traversing a Rooted Plane Tree

As in case of decision trees, we can define the notion of depth first traversals of a rooted plane tree. We can imagine going around the *RP*-tree (Fig. 11.3) following arrows. We can start at the root, 1, go down edge a to vertex 2 etc. We find the sequence of vertices as encountered in this process: 1, 2, 4, 2, 5, 2, 1, 3, 6, 8, 6, 9, 6, 10, 6, 11, 6, 3, 7, 3, 1. This sequence of vertices is known as the depth first vertex sequence, DFV(T),

of the rooted plane Tree *T*. The number of times each vertex appears in *DFV*(*T*) is one plus the number of children of that vertex. For edges, the corresponding sequence is *a*, *c*, *c*, *d*, *d*, *a*, *b*, *e*, *g*, *g*, *h*, *h*, *i*, *i*, *j*, *j*, *e*, *f*, *f*, *b*. This sequence is the depth first edge sequence, *DFE*(*T*), of the tree. Every edge appears exactly twice in *DFE*(*T*). If the vertices of the rooted plane tree are read left to right, top to bottom, we can obtain the sequence 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11. This is called the breadth first vertex sequence, *BFV*(*T*). Similarly, the breadth first edge sequence, *BFE*(*T*) is *a*, *b*, *c*, *d*, *e*, *f*, *g*, *h*, *i*, *j*. (As shown in Fig 11.4).

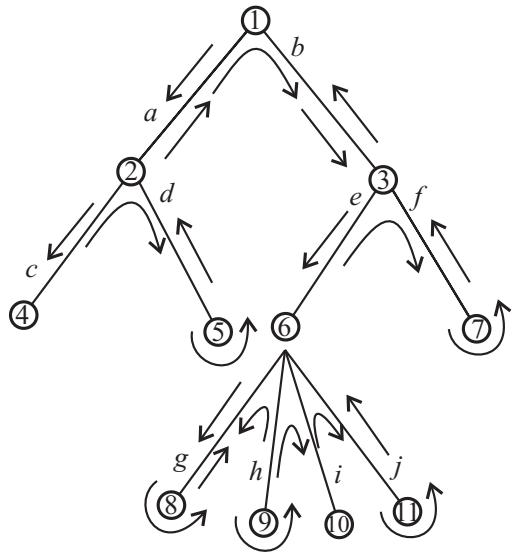


Fig. 11.4: TRP – Tree

The sequence *BFV*(*T*) and *BFE*(*T*) are linear ordering of the vertices and edges of the *RP*-tree *T* (each vertex or edge appears exactly once in the sequence). We also associate linear ordering with *DFV*(*T*) which is called the sequence of vertices of *T*.

PREV(*T*), and the post order sequence of vertices of *T*, *POSV*(*T*).

Here *PREV*(*T*) = 1, 2, 4, 5, 3, 6, 8, 9, 10, 11, 7 is the sequence of first occurrances of the vertices of *T* in *DFV*(*T*).

POSV(*T*) = 4, 5, 2, 8, 9, 10, 11, 6, 7, 3, 1 is the sequence of last occurrences of the vertices of *T* in *DFV*(*T*).

We can notice that the order in which the leaves of *T* appear, 4, 5, 8, 9, 10, 11, is the same in both *PREV*(*T*), and *POSV*(*T*). ■

Theorem 2: A (p, q) graph is a tree iff it is acyclic and $p = q + 1$ or $q = p - 1$.

Proof: If G is a tree, then it is acyclic. From definition, to verify the equation $p = q + 1$.

We can employ inductions on p.

for $p = 1$, the result is so trivial.

We can assume, then that the equality $p = q + 1$ holds for all (p, q) trees with $p \geq 1$ vertices

Let G_1 be a tree with $p + 1$ vertices

Let v be an end vertex of G_1.

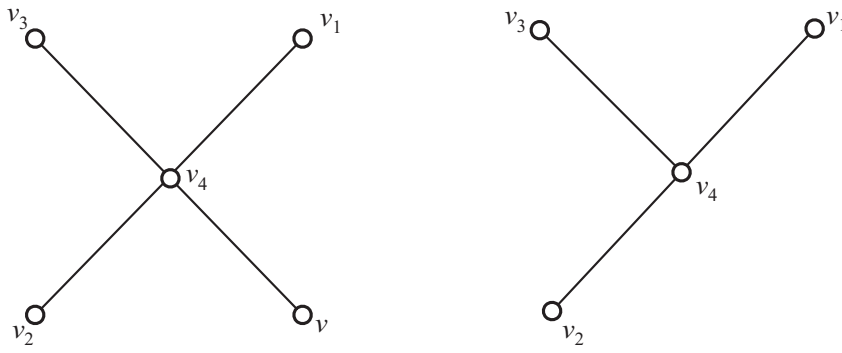
The graph $G_2 = G_1 - v$ is a tree of order p, and so

$$p = |E(G_2)| + 1$$

Since G_1 has one more vertex and one more edge than that of G_2.

$$|V(G_1)| = p + 1 = |E(G_2)| + 1 + 1 = |E(G_1)| + 1.$$

$\therefore$ $|V(G_1)| = |E(G_1)| + 1$. (As shown in Fig. 11.5)



(G_1) $(G_2 = G_1 - v)$

Fig. 11.5

Converse: Let G be an acyclic (p, q) graph with $p = q + 1$

We can show G is a tree. To do so, we need only to verify that G is connected. We can denote by $G_1, G_2,...,G_k$ the components of G, where $k \geq 1$.

Further, Let G_i be a (p_i, q_i) graph

Since each G_i is a tree, $p_i = q_i + 1$.

Hence $$p - 1 = q = \sum_{i=1}^{k} q_i$$

$$= \sum_{i=1}^{k} q_i (p_i - 1) = p - k$$

$\Rightarrow$ $$p - 1 = p - k$$

$\Rightarrow$ $k = 1$. and G is connected.

Hence, (p, q) graph is tree. ■

Theorem 3: A (p, q) graph G is a tree iff G is connected and $p = q + 1$.

Proof: Let G be a (p, q) tree. From definition of G, it is connected and $A(p, q)$ graph is a tree iff it is a acyclic and $p = q + 1$, or $q = p - 1$ (Theorem 2).

Conversely, we assume G is connected (p, q) graphs with $p = q + 1$

It is sufficient to show that G is acyclic.

If G contains a cycle C and e is an edge of C, then $G - e$ is a connected graphs with p vertices having $p - 2$ edges.

It is impossible by definition and contridicts our assumption.

Hence G is connected. ■

Example 4

A tree has five vertices of degree 2, three vertices of degree 3 and four vertices of degree 4. How many vertices of degree 1 does it have?

Solution

Let x be the number of nodes of degree 1.

$\therefore$ Total number of vertices $= 5 + 3 + 4 + x$

$= 12 + x.$

The total degree of tree $= 5 \times 2 + 3 \times 3 + 4 \times 4 + x$

$= 35 + x$

The number of edge in the tree is half of the total degree of it.

If $G = (V, E)$ be the tree, then, we have

$$|V| = 12 + x \text{ and } |E| = \frac{35 + x}{2}$$

If any tree $|E| = |V| - 1$.

$$\Rightarrow \qquad \frac{35+x}{2} = 12 + x - 1$$

$$\Rightarrow \qquad \frac{35+x}{2} = 24 + 2x - 2$$

$$\Rightarrow \qquad x = 13.$$

$\therefore$ There are 13 nodes of degree one in the tree. ■

Example 5

The Number of Labeled Trees

How many n-vertex labeled trees are there? Or, count the number of trees with vertex set $V = \underline{n}$. The answer has been obtained in a variety of ways.

Solution

Suppose f is a function from V to V. We can represent this as a simple digraph (V, E) where the edges are $\{(v, f(v)) \mid v \in V\}$. The function

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\ 1 & 10 & 9 & 2 & 8 & 2 & 2 & 5 & 1 & 6 & 11 \end{pmatrix}$$

corresponds to the directed graph

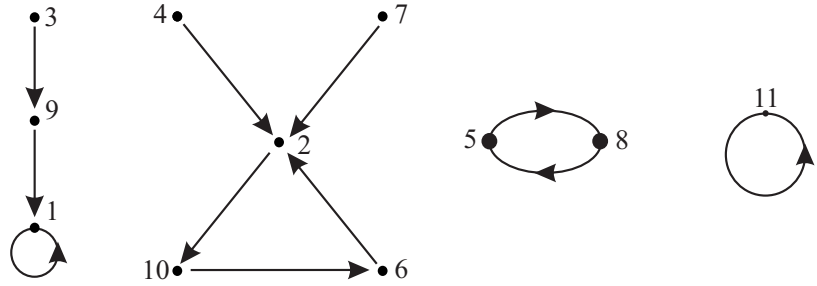


Fig. 11.6: Functional Digraph

Such graphs are called *functional digraphs.* We should be able to convince ourself that a functional digraph consists of cycles (including loops) with each vertex on a cycle being the root of a tree of noncyclic edges. The edges of the trees are directed toward the roots. In the Fig. 11.6.

- 1 is the root of the tree with vertex set {1, 3, 9},

- 2 is the root of the tree with vertex set {2, 4, 7}
- 5 is the root of the tree with vertex set {5},
- 6 is the root of the tree with vertex set {6},
- 8 is the root of the tree with vertex set {8}
- 10 is the root of the tree with vertex set {10} and
- 11 is the root of the tree with vertex set {11}.

In a tree, there is a unique path from the vertex 1 to the vertex n. Remove all the edges on the path and list the vertices on the path, excluding 1 and n, in the order they are encountered. Interpret this list as a permutation in 1 line form. Draw the functional digraph for the cycle form, adding the cycles form, adding the cycles (1) and (n). Add the trees that are attached to each of the cycle vertices, directing their edges toward the cycle vertices. Consider the following Fig. 11.7.

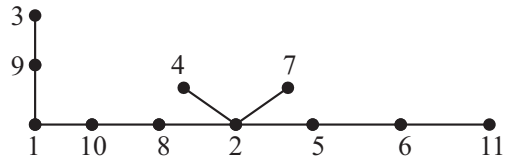


Fig. 11.7 Tree with Cycle Certices

The one line form is 10, 8, 2, 5, 6. In two line form it is. Thus the cycle form is (2, 10, 6) (5, 8). When we add the two cycles (1) and (11) to this, draw the directed graph, and attach the directed trees, we obtain the functional digraph pictured earlier.

This gives us a one-to-one correspondence between trees with $V = \underline{n}$ and functions $f : \underline{n} \to \underline{n}$ with $f(1) = 1$ and $f(n) = n$. In creating such a function, there are n choices for each of $f(2)$, ..., $f(n - 1)$. Thus there are n^{n-2} such functions and hence n^{n-2} trees. ■

Example 6

A tree has 2n vertices of degree 1, 3n vertices of degree 2 and n vertices of degree 3. Determine the number of vertices and edges in the tree.

Solution

We are given that total number of vertices in the tree is $2n + 3n + n = 6n$

The total degree of the tree is $2n \times 1 + 3n \times 2 + n \times 3 = 11n$.

The number of edges in the tree will be half of $11n$.

If $G = (V, E)$ be a tree then, we have

$$|V| = 6n \text{ and } |E| = \frac{11n}{2}$$

for a tree $|E| = |V| - 1$

We have, $\frac{11n}{2} = 6n - 1.$

$\Rightarrow$ $11n = 12n - 2$

$\Rightarrow$ $n = 2$

$\Rightarrow$ nodes = 12, edges = 11 ■

11.7 Binary Trees

A binary tree is a rooted tree where each vertex v has atmost two subtrees; if both subtrees are present, one is called a left subtree of v and the other is called right subtree of u. Iff only one subtree is present, it can be designated either as the left subtree or right subtree of v.

OR

A binary tree is a 2-ary tree in which each child is designated as a left child or right child.

In a binary tree every vertex has two children on no children.

A binary tree has the following properties:

(*i*) The number of vertices n in a complete binary tree is always odd. This is because there is exactly one vertex of even degree, and remaining $(n - 1)$ vertices are of odd degree. Since from the theorem (number of vertices of odd degree is even) $n - 1$ is even. Hence n is odd.

(*ii*) Let p be the number of end vertices in a binary tree T. Then $n - p - 1$ is the number of vertices of degree 3. The number of edges in T is

$$\frac{1}{2}[p + 3(n - p - 1) + 2] = n - 1.$$

on $$p = \frac{n+1}{2}$$

(*iii*) A non-end vertex in a binary tree is called an internal vertex. The number of internal vertices in a binary tree is one less than the number of end vertices.

(*iv*) In a binary tree, a vertex v_i is said to be at level l_i and v_i is at a distance l_i from the root. Thus the root is at level *O*.

(*v*) The maximum number of vertices possible in a k = level binary tree is $2^0 + 2^1 + 2^2 + 2^3 + ...+ 2^k \geq n$.

(*vi*) The maximum level I_{max} of any vertex in a binary tree is called the *height* of the tree.

(*vii*) To construct a binary tree for a given n s.t. the farthest vertex is as far as possible from the root, we must have exactly two vertices at each level, except at the O level.

Hence max. $I_{max.} = \frac{n-1}{2}$.

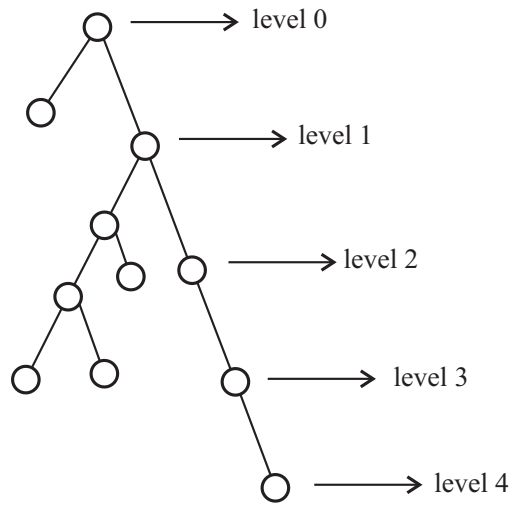


Fig. 11.8: 13-vertices, 4-level Binary Tree

(*viii*) The minimum possible height of n-vertex binary tree is min $I_{max} = [\log_2 (n + 1) - 1]$

(*ix*) The sum of levels of all end vertices is called *path length* of a tree.

Theorem 4: There are at the most n^h leaves in an n-ary tree of height h.

Proof: The theorem can be proved by the principle of mathematical induction on the height of the tree.

As basic step tobe $h = 0$ *i.e.* tree consists of root node only.

Since $n^0 = 1$, the basis step is tree.

We assume that the above statement is tree for $h = k$.

i.e. an n-ary tree of height k has at the most n^k leaves.

If we add n nodes to each of the leaf node of n-ary tree of height k, the total number of leaf nodes will be the most

$$n^h \times n = n^{h+1}$$

Hence inductive step is also true

This proves the statement for all $h \geq 0$. ■

Theorem 5: In a complete n-ary tree with m internal nodes, the number of leaf node l is given by the formula $l = \dfrac{(n-1)(x-1)}{n}$

Where x is the total number of nodes in the tree.

Proof: Given that, the tree has m internal nodes and it is complete n-ary, so total number of nodes.

$$x = n \times m + 1.$$

Then, we have, $$m = \frac{x-1}{n}$$

It is also given that l is the number of leaf nodes in the tree. We have, $x = m + l + 1$.

Substituting the value of m in this equation, we get

$$x = \left(\frac{x-1}{n}\right) + (l+1)$$

$$\Rightarrow \qquad l = \frac{(n-1)(x-1)}{n}$$ ■

11.8 Spanning Trees

A spanning tree of a simple graph $G = (V, E)$ is a subgraph $T = (V, E')$ which is a tree and has the same set of vertices as G.

Connected graphs and Spanning Trees

Since a tree is connected, a graph with a spanning tree must be connected. On the other hand, it is not hard to see that every connected graph has a spanning tree. Any simple graph $G = (V, E)$ has a subgraph that is a tree, $T' = (V', E')$. Take $V' = \{v\}$ to be one vertex and E' empty. Suppose that $T' = (V' , E')$ is the largest such "subtree". If T' is not a

spanning tree then there is a vertex w of G that is not a vertex of T'. If G is connected. We choose a vertex u in T' and a path $w = x_1, x_2,..., e_k = u$ from w to u. Let j, $1 < j \leq k$, be the first integer such that x_j is a vertex of T'. Then adding the edge $\{x_j - 1, x_j\}$ and the vertex x_{j-1} to T' creates a subtree T of G that is larger than T', a contradiction of the maximality of T'. We have, in fact, shown that a graph is connected if and only if every maximal subtree is a spanning tree. Thus we have: *A graph is connected if and only if it has a spanning tree.* It follows that, if we had an algorithm that was guaranteed to find a spanning tree whenever such a tree exists, then this algorithm could be used to decide if a graph is connected.

Weight in a Graph

Let $G = (V, E)$ be a simple graph and let λ (e) a function from E to the positive real numbers. We call $\lambda(e)$ the weight of the edge. If $H = (V', E')$ is a subgraph of G, then $\lambda(H)$, the weight of H, is the sum of $\lambda(e')$ over all $e' \in E'$.

A minimum weight spanning tree for a connected graph G is a spanning tree such that $\lambda(T) \leq \lambda(T')$ whenever T' is another spanning tree.

Theorem 6: Each connected graph has a spanning tree, *i.e.,* a spanning graph is a tree.

Proof: Let $H \subseteq G$ be a minimal connected spanning subgraph, *i.e.* a connected spanning subgraph of G s.t. $H - e$ is disconnected for all $e \in E_H$. Such a subgraph is obtained from G by removing nonbridges.

- To start with, let $H_0 = G$.
- For $i \geq$ o, let $H_{i+1} = H_i - e_i$ where e_i is not a bridge of H_i. Since e_i is not a bridge H_{i+1} is a connected spanning subgraph of H_i and thus of G.
- $H = H_k$ when only bridges are left.

Hence H is a tree. ■

Theorem 7: A nondirected graphs G is connected iff G containing a spanning tree. Indeed, if we successive –ly delete edges of cycles until no further cycles remains, then the result is a spanning tree of G.

Proof: If G has a spanning tree T, there is a path between any pair of vertices in G along the tree T.

Thus G is connected.

Conversely, We can prove that a connected graph G has a spanning tree by *PMI* on the number k of cycles in G. If $k = 0$, then G is connected

with no cycles and hence G is already a tree. Suppose that all connected graphs with fewer than k cycles have a spanning tree. Then suppose that G is a connected graph with k cycles. Remove an edge e from one of the cycles. Then $G - e$ is still connected and has a spanning tree by the inductive hypothesis because $G - e$ has fewer cycles then G. But since $G - e$ has all the vertices of G, the spanning tree for $G - e$ is also one for G. ■

11.9 Breadth – First Search and Depth – First Search (BFS and DFS)

An algorithm based on the proof of theorem 7 could be designed to produce a spanning tree for a connected graph. If we recall from the proof that all that one need is destroy cycles in the graph by removing an edge from a cycle until no cycles remain. Unfortunately, such an algorithm not very efficient because it consumes a long time to find cycles. On the other hand, we can define other rather efficient algorithms for finding a spanning tree of a connected graphs. There algorithms are called breadth – first search (BFS) and depth – first search (DFS).

BSF Algorithm for a spanning tree

Input: A connected graph G with Vertices $v_1, v_2,...,v_n$.

Output: A spanning tree T for G.

Method

(*i*) (Start.) Let v_1 be the root of T. From the set $V = \{v_1\}$.

(*ii*) (Add new edges.) Consider the vertices of V in order consistent with the original labeling. Then for each vertex $x \in V$, add the edge $\{x, v_k\}$. to T where k is the minimum index such that adding the edge $\{x, v_k)$. to T does not produce a cycle. If no edge can be added, then stop; T is a spanning tree for G. After all the vertices of V have been considered in order, go to step (*iii*)

(*iii*) (Update V.) Replace V by all the children v in T of the vertices x of V where the edges $\{x, v\}$ were added in step (*ii*). Go back and repeat step (*ii*) for the new set V.

General Procedure

(*i*) Arbitrarily choose a vertex and designate it as the root. Then add all edges incident to this vertex, such that the addition of edges does not produce any cycle.

(*ii*) The new vertices added at this stage become the vertices at level 1 in a spanning tree, arbitrarily order them.

(*iii*) Next, for each vertex at level 1, visited in order, add each edge incident to this vertex to the tree as long as it does not produce any cycle.

(*iv*) Arbitrarily order the children of each vertex at level 1. This produces the vertices at level 2 in the tree.

(*v*) Continue the same procedure until all the vertices in the tree have been added.

(*vi*) The procedure ends, since there are only a finite number of edges in the graph.

(*vii*) A spanning tree is produced since we have already produced a tree without cycle containing every vertex of the graph.

Example 7

Use BFS algorithm to find a spanning tree of graph G of Fig. 11.9.

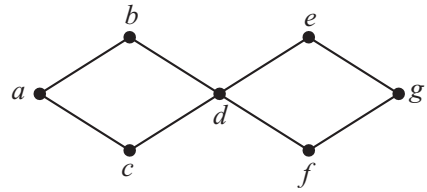


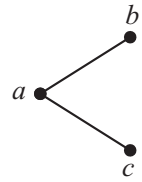
Fig. 11.9

Solution

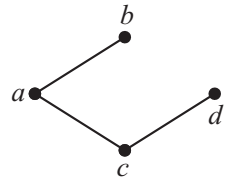
Step 1: Choose the vertex *a* as the root

• *a*

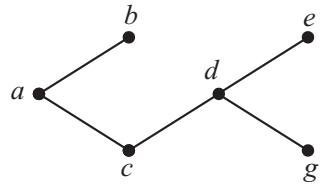
Step 2: Add edges incident with all vertices adjacent to *a*, so that edges {*a*, *b*}, {*a*, *c*} are added. The two vertices *b* and *c* are in level 1 in the tree.



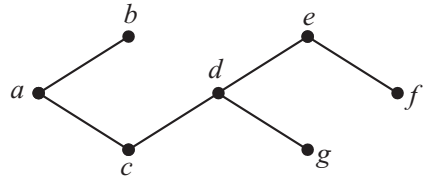
Step 3: Add edges from these vertices at level 1 to adjacent vertices not already in the tree. Hence the edge $\{c, d\}$ is to be added. The vertex d is in level 2.



Step 4: Add edge from d in level 2 to adjacent vertices not already in the tree. The edge $\{d, e\}$ and $\{d, g\}$ are added. e and g are is level 3.



Step 5: Add edge from e at level 3 to adjacent vertices not already in the tree and hence $\{e, f\}$ is added. The steps of Breath first procedure are obtained.



Example 8

Illustrate BFS on the graph given in Fig. 11.10.

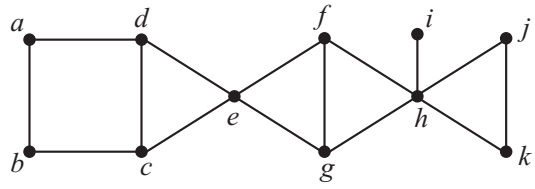


Fig. 11.10

Solution

We firstly select the ordering of the vertices $a, b, c, d, e, f, g, h, i, j, k$.

(*i*) Select a as the first vertex in the spanning tree T and designate it as the root of T. At this stage, T consists of the single vertex a. Add to T all edges $\{a, x\}$ as x sums in order from b to k, that do not produce a cycle in T.

(*ii*) Add $\{a, b\}$ and $\{a, d\}$. These edges are called tree edges for the BFS.

(*iii*) Repeat the process for all vertices on level one from the root by examining each vertex in the designated order. Since b and d are at level 1, we first examine b.

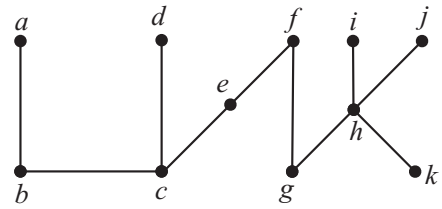
(*iv*) For bc, we include the edge $\{b, c\}$ as a tree edge. For d, we reject the edge $\{d, c\}$ since inclusion would produce a cycle in T. But we include $\{d, e\}$.

(*v*) Consider the vertices at level two. Reject the edge $\{c, e\}$; include $\{e, f\}$ and $\{e, g\}$.

(*vi*) Repeat the procedure again for vertices on level three. Reject the $\{f, g\}$, but include $\{f, h\}$. At g, reject $\{f, g\}$ and $\{g, h\}$.

(*vii*) On level four, include $\{h, i\}$, $\{h, j\}$, $\{h, k\}$.

(*viii*) We attempt to apply the procedure on level five at i, j and k, but no edge can be added at these vertices so the procedure ends.



■

DFS Algorithm for Spanning Tree

Input: A connected graph G with vertices $v_1, v_2, ..., v_n$.

Output: A spanning tree T for G.

Method

(*i*) (Visit a vertex). Let v_1 be the root of T, and set $L - v_1$ (L is the vertex last visited)

(*ii*) (Find an unexamined edge and an unvisited vertex adjacent to L) For all vertices adjacent to L, choose the edge $\{L, V_k\}$, where k is the minimum index such that adding $\{L, v_k\}$ to T does not create a cycle. If no such edge exists, go to step (*iii*) otherwise, add edge $\{L, v_k\}$ to T and set $L - v_k$ *i* repeat step (*ii*) at the new value for L.

(*iii*) (Back track or terminate). If x is the parent of L in T, set $(L - x)$ and apply step (*ii*) at the new value of L. If, L has no parent in T (so that $L - v_1$) then the DFS terminates and T is a spanning tree for G.

General Procedure

(*i*) Arbitrarily choose a vertex from the vertices of the graph and designate it as the root.

(*ii*) From a path starting at this vertex by successively adding edges as long as possible where each new edge is incident with the least vertex in the path without producing any cycle.

(*iii*) If the path goes through all vertices of the graph, the tree consisting of this path is a spanning tree. Otherwise, move back to the next to last vertex in the path, and, if possible, form a new path starting at this vertex passing through vertices that were not previously visited.

(*iv*) If this cannot be done, move back another vertex in the path, that is two vertices back in the path and repeat.

(*v*) Repeat this procedure, beginning at the last vertex visited, moving back up and path one vertex at a time, forming new paths that are as long as possible until no more edges can be added.

(*vi*) This process ends since the graph has a finite number of edges and is connected. A spanning tree is produced.

Example 9

Use DFS to construct a spanning tree of the graph given in Fig. 11.11.

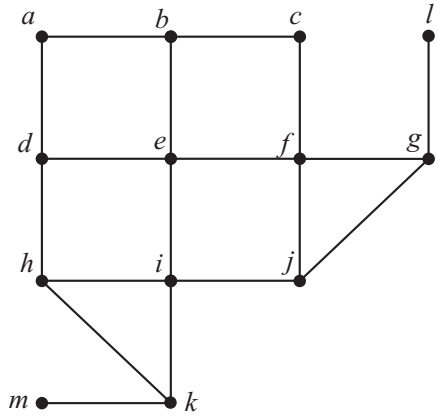
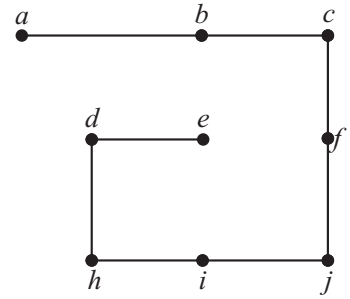


Fig. 11.11

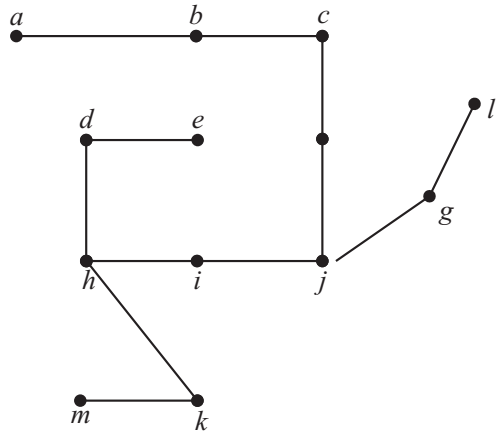
Solution



(*i*) Start with vertex *a*, build a path by successively adding edges incident with vertices not already in the path, as long as this is possible.

This produces a path $a - b - e - f - g - i - h - d - e$.

(*ii*) Now back track to *d*. There is no path at *d* containing vertices no already visited. So move back track to *h* and form the path $h - k - m$. Now back track to *k*, and *h*, and *i* and *j* then form the path $j - g - l$. This produces the spanning tree.



■

Example 10

Find a spanning tree of the graph of Fig. 11.12 using DFS algorithm.

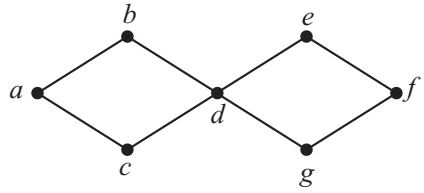


Fig. 11.12

Solution

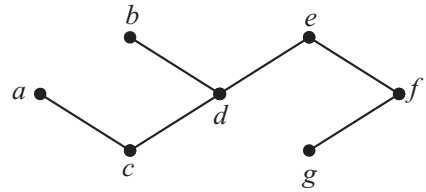
(*i*) Choose the vertex *a*.

(*ii*) Form a path by successively adding edges incident with vertices not already in the path as long as possible.

(*iii*) Now back track of *F*. There is no path beginning at *F* containing vertices not already visited.

(*iv*) Similarly, after backtrack at *e*, there is no path. So, move back track at *d* and form the path *d* – *b*.

This produces the required spanning tree which is shown in Fig. below



■

Theorem 8: For every tree $T = (V, E)$, if $|V| \geq 2$, then T has atleast two pendent vertices

Proof: Let $|V| = n \geq 2$

Same $|E| = n - 1$, so, if $G = (V, E)$ is an undirected graph or multigraph then $\sum_{v \in V} \deg(v) = 2|E|$

It follows that $2(n-1) = 2|E| = \sum_{v \in V} \deg(v)$

Since *T* is connected, we have $\deg(v) \geq 1 \ \forall \ v \in V$.

If T has fewer than two pendent vertices, then either

$$\deg(v) \geq v \in V$$

$$\deg(v^*) = 1 \text{ for only once vertex } v^* \text{ in } V.$$

In the first case we arrive at the contradictions.

$$2(n-1) = \sum_{v \in V} \deg(v) \geq 2\,|V| = 2n$$

for second case we find that

$$2\,(n-1) = \sum_{v \in V} \deg(v) \geq 1 + 2(n-1).$$

Which is another contradiction ■

Example 11

Let $T = (V, E)$ be a tree with $V = \{v_1, v_2, ..., v_n)$ for $n \geq 2$. Prove that the number of pendent vertices in T is equal to $2 + \sum_{deg(v_i) \geq 3} \left[deg(v_1) - 2 \right]$

Solution

Let $1 \leq i\ (< n)$. Let x_i = no. of vertices v_1 where $\deg(v) = i$

Then $x_1 + x_2 +, ..., + x_{n-1} = |V| = |E| + 1.$

So, $2\,|E| = 2\,(-1 + x_1 + x_2 +, ... + x_{n-1})$

But $$2|E| = \sum_{v \in V} \deg(v)$$

$$= \{(x_1 + 2x_2 + 3x_3 + ... + (n-1)\,x_{n-1}\}$$

Solving, we get

$$2\,(-1 + x_1 + x_2 + ... + x_{n-1}) = x_1 + 2x_2 + 3x_3 + ... + (n-1)\,x_{n-1}$$

for x_1, we find that

$$x_1 = 2 + x_3 + 2x_4 + 3x_5 + ... + (n-3)\,x_{n-1}$$

$$= 2 + \sum_{\deg(v_i) \geq 3} \left[\deg(v_i) - 2 \right]$$ ■

11.10 Minimal Spanning Trees

The application of spanning trees are many and varied, and in order to gain some appreciation for this fact, we would describe what is sometimes called the connector problem. Let us consider that we have a collection of n cities, and that we wish to construct a utility, communication, or transportation network connecting all of the cities. Assuming that we know the cost of building the links between each pair of cities and that, in addition, we wish to construct the network as cheaply as possible. The desired network can be represented by a graph by regarding each city as a vertex and by placing an edge between vertices if a link runs between the two corresponding cities.

Let G be the graph of all possible links between the cities with the non-negative cost of construction $c(e)$ assigned to each edge e in G. Then if H is any subgraph of G with edges $e_1, e_2,...,e_m$ the total cost of constructing the network H is

$$C(H) = \sum_{i-1}^{m} C(e_i)$$

A spanning tree T where $c(T)$ is minimal is called a minimal spanning tree of G.

11.10.1 Kruskal's Algorithm (for Finding a Minimal Spanning Tree)

Input: A connected graph G with non negative values assigned to each edge.

Output: A minimal spanning tree for G.

Step 1: Select any edge of minimal value that is not a loop. This is the first edge of T.

(If there is more than one edge of minimal value, arbitrarily choose one of these edges).

Step 2: Select any remaining edge of G having minimal value that does not for a circuit with the edges already included in T.

Step 3: Continue step 2 until T contains $n - 1$ edges, where n is the number of vertices of G.

Theorem 9: Let G be a connected graph where the edges of G are labelled by nonnegative numbers. Let T be an economy tree of G obtained from Kruskal's Algorithm. Then T is a minimal spanning Tree.

Proof: For each edge e of G, let $C(e)$ be the value assigned to the edges by the labelling.

If G has n vertices, an economy tree T must have $(n-1)$ edges.

Let the edges $e_1, e_2, e_3,...,e_{n-1}$ be chosen in the Krushal's Algorithm.

Then $$C(T) = \sum_{i-1}^{n-1} C(e_i) \qquad \ldots(i)$$

Let T_0 be a minimal spanning tree of G. We can show that $C(T_0) = C(T)$, and conclude that T is also minimal spanning tree.

If T and T_0 are not the same, let e_i be the first edge of T not in T_0.

Adding the edge e_i to T_0 we obtain the graph G_0.

Suppose $e_i = \{a, b\}$. Then a path P from a to b exists in T_0 and so P together with e_i produces a circuit C in G_0.

Since T contains no circuits, there must be an edge e_0 in C that is not in T.

The graph $T_1 = G_0 - e_0$ is also a spanning tree of G since T_1 has $(n-1)$ edges.

Moreover,

$$C(T_1) = C(T_0) + C(e_i) - C(e_0)$$

We know that, $C(T_0) \le C(T_1)$ since T_0 was a minimal spanning tree of G.

Thus

$$C(T_1) - C(T_0) = C(e_i) - C(e_0) \ge 0.$$

It implies that, $C(e_i) \ge C(e_0)$.

Since T was constructed by Kruskal's Algorithm, e_i is an edge of smallest value that can be added to the edges $e_1, e_2, e_3,...,e_{i-1}$ without producing a circuit.

Also, if e_0 is added to the edges $e_1, e_2, e_3, ..., e_{i-1}$, no circuit is produced because the graph formed is a subgraph of the tree T_0.

Therefore, $C(e_i) = C(e_0)$,

so that $C(T_1) = C(T_0)$.

We have constructed from T_0 a new minimal spanning tree T_1 such that the number of edges common to T_1 and T exceeds the number of edges common to T_0 and T by one edge, namely e_i.

Repeating this procedure, to construct minimal spanning tree T_2 with one more edge in common with T than was in common between T_1 and T.

By containing this process, we finally reach at a minimal spanning tree with all edges in common with T, and we conclude that T is a minimal spanning tree. ■

Example 12

Use Kruskal's Algorithm to find minimal spanning tree for Fig. 11.13. Also find the cost of it.

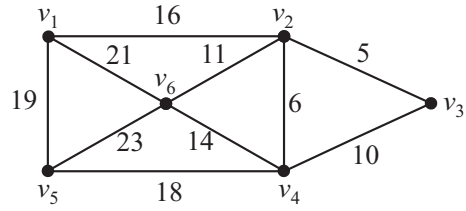


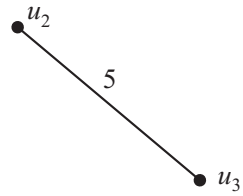
Fig. 11.13

Solution

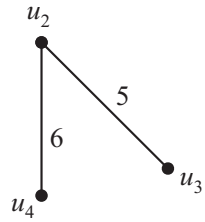
We collect lengths of edges into a table.

Edge	*Cost*
$v_2 - v_3$	5
$v_2 - v_4$	6
$v_4 - v_3$	10
$v_2 - v_6$	11
$v_4 - v_6$	14
$v_2 - v_4$	16
$v_4 - v_5$	18
$v_5 - v_1$	19
$v_1 - v_6$	21
$v_5 - v_6$	23

(*i*) Choose the edge $v_2 - v_3$ (Min. wt.)

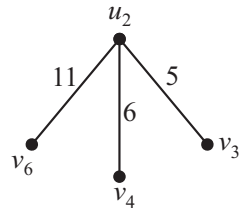


(*ii*) Add the next edge with min. wt.



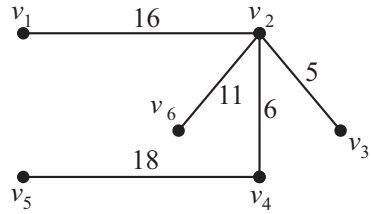
(*iii*) Reject the edge $v_4 - v_3$, because it forms a cycle.

(*iv*) Add the edge $v_2 - v_6$



(*v*) Reject $v_4 - v_6$ since it forms cycle.

(*vi*) Add the edges $v_2 - v_1$, $v_4 - v_5$



Now all the vertices of G are covered, therefore we stop the algorithm.

Cost or weight of Minimal spanning tree

$$= 5 + 6 + 11 + 16 + 18$$

$$= 56.$$

■

Example 13

Determine a railway network of minimal cost for the cities as shown in Fig. 11.14. Also find minimal cost.

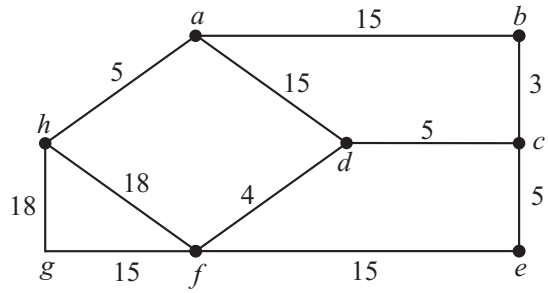


Fig. 11.14

Solution

Collecting lengths of edges into a table, we set:

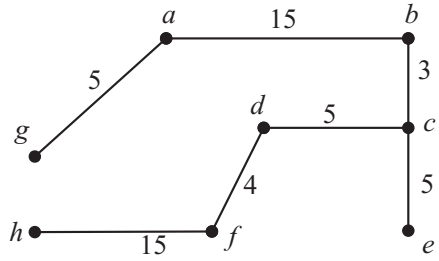
Edge	*Cost*
$b-c$	3
$d-f$	4
$a-g$	5
$c-d$	5
$c-e$	5
$a-b$	15
$a-d$	15
$f-h$	15
$g-h$	15
$e-f$	15
$f-g$	18

(*i*) Choose the edges $b-c$, $d-f$, $a-g$, $c-d$ and $c-e$.

(*ii*) We have options: we may choose only one of $a-b$ and $a-d$ for the selection of both creates a circuit. Suppose we choose $a-b$.

(*iii*) Likewise we may choose only one of $g-b$ and $f-h$. Suppose we choose $f-h$.

(*iv*) We then have a spanning tree as illustrated in the Fig. below.



The minimal cost for construction of this tree is

$$= 3 + 4 + 5 + 5 + 5 + 15 + 15$$

$$= 52$$ ■

11.10.2 Prim's Algorithm

Input: A connected weighted graph G with n vertices.

Output: A minimal spanning tree T.

Step 1: Select an arbitrary vertex v_1 and an edge e_1 with minimum weight incident with vertex v_1. This forms initial MST, T.

Step 2: If edges $e_1, e_2, e_3,...,e_i$ have been chosen involving end points $v_1, v_2, ..., v_{i+1}$. Choose and edge $e_{i+1} = v_j\, v_k$ with $v_j \in T$ and $v_k \in T$ s.f. e_{i+1} has smallest weight among the edges of G with precisely one and in $\{v_i, v_2, ..., v_{i+1}\}$.

Step 3: Stop after $n - 1$ edges have been chosen. Otherwise go to step 2.

Example 14

Find the minimal spanning tree of the weighted graph of Fig. 11.15, using Prim's Algorithm.

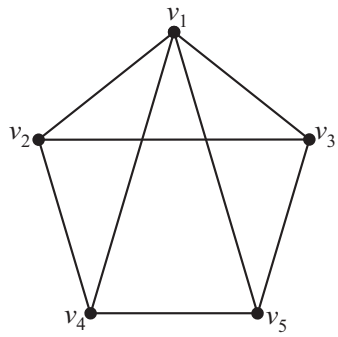
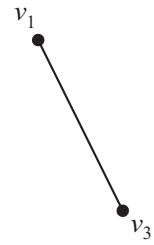


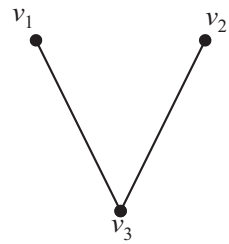
Fig. 11.15

Solution

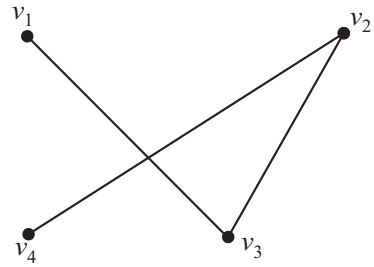
(*i*) We choose the vertex v_1. Now edge with smallest weight incident on v_1, is (v_1, v_3), so we choose the edge on (v_1, v_3).



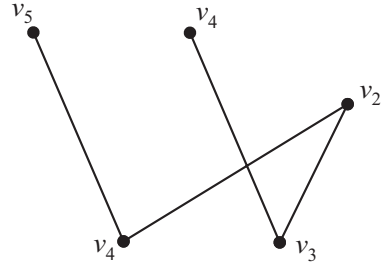
(*ii*) $w(v_1, v_2) = 4$, $w(v_1, v_4) = 3$, $w(v_1, v_5) = 4$, $w(v_1, v_2) = 2$ and $w(v_3, v_4) = 3$. So, we choose the edge (v_3, v_2) since it is of minimum weight.



(*iii*) $w(v_1, v_5) = 3$, $w(v_2, v_4) = 1$. and $w(v_3, v_4) = 3$. We choose the edge (v_2, v_4).



(*iv*) We choose the edge (v_4, v_5). Now all vertices are covered. The minimum spanning tree is produced.



■

Example 15

Find the minimal spanning tree of the weighted graph using Prim's Algorithm.

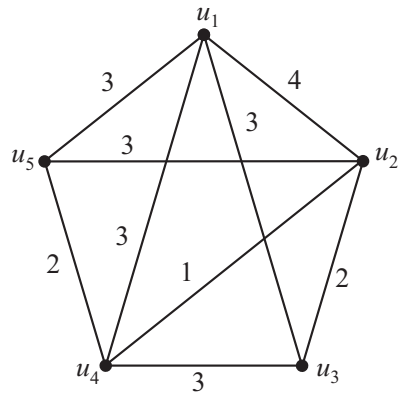


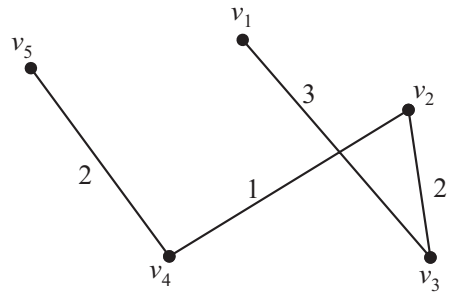
Fig. 11.16

Solution

According to step 1, choose vertex v_1. Now edge with smallest weigh incident on v_1, is $e = v_1\ v_3$ or $v_1\ v_5$, choose $e = v_1\ v_3$.

Similarly choose the edges $v_3\ v_2$, $v_2\ v_4$, $v_4\ v_5$

The minimal spanning tree is



■

11.11 Directed Trees

Two vertices u and v of a directed graph G are called quasi – strongly connected if there is a vertex w from which there is a direct path to u and a directed path to v. If there is a directed path P from u to v then certainly u and v are quasi – strongly connected, because we can take w to be u

itself, and then there is a trivial path with no edges from u to v and the path P from u to v.

The graph G is said to be quasi – strongly connected if each pair of vertices of G is quasi – strongly connected. It is clear that if a directed graph G is quasi – strongly connected, then the underlying non directed graphs will be connected. The digraph (Fig. 11.17) is quasi – Strongly connected.

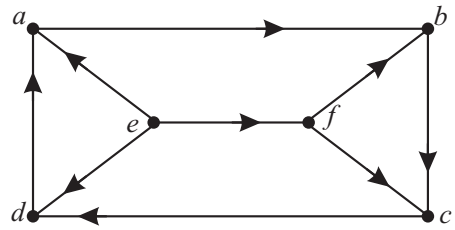


Fig. 11.17: Quasi-strongly Connected Graph

A digraph a has a directed spanning tree iff G is quasi – strongly connected.

A directed forest is a collection of directed trees. The height of vertex v in a directed forest in the length of the longest directed path from v to a leaf. The height of a nonempty tree is the height of its root.

The level of a vertex v in a forest is the length of the path to v from the root of the tree to which it belongs. A directed tree T is said to have degree k if k is the maximum of the out-degrees of all the vertices in T.

Theorem 10: Let G be a digraph. Then the following are equivalent:

(i) G is quasi – strongly connected.

(ii) There is a vertex r in G such that there is a directed path from r to all the remaining vertices of G.

Proof: It is clear that (ii) $\Rightarrow$ (i)

On the other hand, Let G is quasi – strongly connected and we consider its vertices $v_1, v_2, ..., v_n$. There is a vertex w_2 from which there is a path to v_1 and a path to v_2. There might also be a vertex w_3 from which there is a path to w_2 and w_3, and so on finally we conclude that there is a vertex w_n from which there is a path to w_{n-1} and a path to v_n.

It is clear that, there is a directed path from w_n to each vertex v_i of G since w_n is connected to $v_1, ..., v_{n-1}$ through w_{n-1}. Hence (i) $\Rightarrow$ (ii). ■

SOLVED EXAMPLES

Example 16

Suppose that a tree T has N_1 vertices of degree 1. N_2 vertices of degree 2, N_3 vertices of degree 3, ..., N_k vertices of degree k. Prove that

$$N_1 = 2 + N_3 + 2N_4 + 3N_5 + ... + (k - 2)\, N_k.$$

Solution

We consider a tree T.

The total number of vertices $= N_1 + N_2 + N_3 + ... + N_k$

Sum of degrees of vertices $= N_1 + 2N_2 + 3N_3 + ... + N_k$.

$\therefore$ Total number of edges in $T = N_1 + N_2 + ... + N_k - 1$.

by handshaking property, we have

$$N_1 + 2N_2 + 3N_3 + ... + kN_k = 2(N_1 + N_2 + ... + N_k - 1)$$

Solving, we get

$$N_3 + 2N_4 + 3N_5 + ... + (k - 2)\, N_k = N_1 - 2$$

$$\Rightarrow \qquad N_1 = 2 + N_3 + 2N_4 + 3N_5 + ... + (k - 2)\, N_k.$$

■

Example 17

Which trees are complete bipartite graphs?

Solution

Let T be a tree, which is a complete bipartite graph.

Let $T = K_{m, n}$ then the number of vertices in $T = m + n$

$\therefore$ The tree contains $(m + n - 1)$ number of edges.

But the graph $k_{m, n}$ number of edges.

Therefore, $\qquad m + n - 1 = mn.$

$$\Rightarrow \qquad mn - m - n + 1 = 0$$

$$\Rightarrow \qquad m(n - 1) - 1(n - 1) = 0$$

$$\Rightarrow \qquad (m - 1)\,(n - 1) = 0$$

$$\Rightarrow \qquad m = 1 \text{ or } n = 1.$$

This means that T is either $k_{1,n}$ or $k_{m,1}$.

i.e. T is a star.

Example 18

Let $G = (V, E)$ be a loop free undirected graph with $|V| = n$. Prove that G is a tree if and only if, $P(G, \lambda) = \lambda(\lambda - 1)^{n-1}$.

Solution

If G is a tree. We consider G is a rooted tree. There are λ choices for closing the root of G and $(\lambda - 1)$ choices for colouring each of its descendents. The result followed by rule of product conversely, if $P(G, \lambda) = \lambda(\lambda - 1)^{n-1}$, then since the factor λ occurs only once, the graph G is connected. $P(G, \lambda) = \lambda(\lambda - 1)^{n-1}$, then since the factor λ occurs only once, the graph G is connected.

$$P(G, \lambda) = \lambda(\lambda - 1)^{n-1} = \lambda^n - (n-1)\lambda^{n-1} + \ldots + (\lambda)^{n-1}\lambda.$$

$\Rightarrow G$ has n vertices and $(n - 1)$ edges.

Hence G is a tree. ■

Example 19

Using the Kruskal's Algorithm, find a minimal spanning tree for the weighted graph given below (Fig. 11.18).

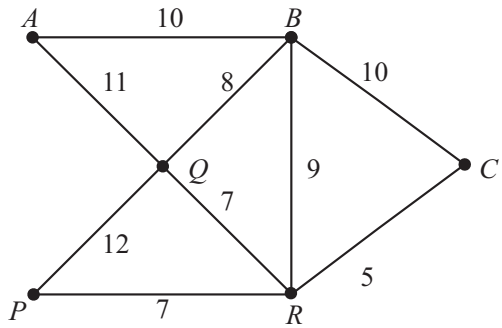


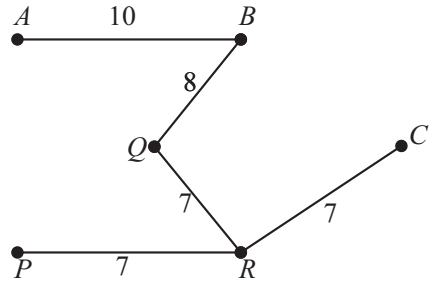
Fig. 11.18

Solution

The given graph has 6 vertices hence a spanning tree will have $(6 - 1) = 5$ edges.

Edge	*Weight*	*Select*
$C - R$	5	✓
$P - R$	7	✓
$Q - R$	7	✓
$B - Q$	8	✓
$B - R$	9	×
$A - B$	10	✓
$B - C$	10	×
$A - R$	11	×
$P - Q$	12	×

A minimal spanning tree of given graph contains the edges $C - R$, $P - R$, $Q - R$, $B - Q$, $A - B$. The tree will be

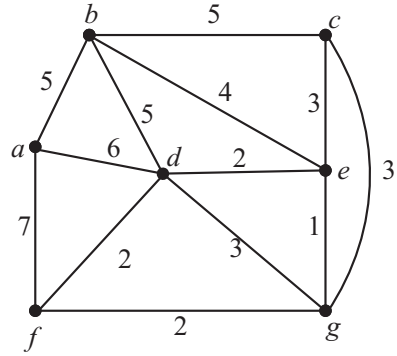


The weight of the tree will be

$$5 + 7 + 7 + 8 + 10 = 37 \text{ units.}$$ ■

Example 20

Use Prim's algorithm find an optimal graph for the graph in Fig. 11.19.



Solution

An optimal tree can be generated as follows:

To initialize: $i = 1, P = \{a\}, N = \{b, c, d, e, f, g\}, T = \phi$.

I iteration: $T = [\{a, b\}], P = \{a, b\}, N = \{c, d, e, f, g\}, i = 2$.

II iteration: $T = [\{a, b\}, \{b, e\}], P = \{a, b, e\},$
$N = \{c, d, f, g\}, i = 3$.

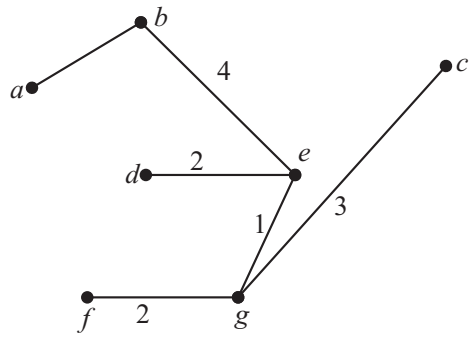
III iteration: $T = [\{a, b\}, \{b, e\}, \{e, g\}], P = \{a, b, e, g\},$
$N = \{c, d, f\}, i = 4$

IV iteration: $T = [\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}, \{f, g\}]$

V iteration: $T = [\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}, \{f, g\}]$
$P = \{a, b, e, g, d, f\}, N = \{c\}, i = 6$.

VI iteration: $T = [\{a, b\}, \{b, e\}, \{e, g\}, \{d, e\}, \{f, g\}, \{c, g\}]$
$P = \{a, b, e, g, d, f, c\} = V, n = \phi, i = 7 = |V|$

Hence it can be proved that T is an optimal spanning tree of weight 17 for G.



of $T = 1 + 2 + 2 + 3 + 4 + 5 = 17$. ■

SUMMARY

1. A **tree** is a connected acyclic graph.
2. A connected graph is a tree if and only if a unique, simple path runs between any two vertices.
3. A connected graph with n vertices is a tree if and only if it has exactly $n - 1$ edges.

4. A **spanning tree** of a connected graph contains every vertex of the graph.
5. Every connected graph has a spanning tree.
6. **Kruskal's algorithm**, the **DFS method**, and **the BFS method** and find spanning trees.
7. A **minimal spanning tree** of a connected weighted graph weighs the least.
8. **Kruskal's algorithm** and **Prim's algorithm** can find minimal spanning trees.
9. A specially designated vertex in a tree is the root of the tree. A tree with a root is **rooted tree**.
10. The **subtree** rooted at v consists of v, its descendants, and the edges incident with them.
11. The **level** of a vertex is the length of the path from the root to the vertex.
12. The **height** of a tree is the maximum level of any leaf in the tree.
13. In an **ordered rooted tree** the children of every vertex are ordered.
14. An m- ary ***tree*** is a rooted tree in which every vertex has at most m children. It is **binary** if $m = 2$, and **ternary** if $m = 3$.
15. An m-ary tree is **full** if every internal vertex has m children.
16. An m-ary tree is **balanced** if all leaves fall on the same level or two adjacent levels.
17. An m-ary tree is **complete** if all leaves lie at the same level.
18. An m-ary tree of height h has at most m^h leaves.
19. For an m-ary tree of height h with l leaves, $h \geq [\log_m l]$.
20. If it is full and balanced, $h = [\log_m l]$.
21. **Preorder, inorder,** and **postorder traversal** are three ways to visit every vertex of a binary tree.
22. An algebraic expression with only binary operators can be written in **prefix, infix**, or **postfix form**. In prefix form, each operator precedes its operands. The other two forms behave similarly.
23. An algebraic expression containing only binary operators can be represented by a **binary expression tree**.

24. A **binary search tree** is homogeneous with every element in the left subtree of every vertex v less than v and every right subtree element is greater than v.
25. A **Huffman code,** a variable-length code, minimize the length of encoded messages.
26. A **decision tree** is an m-ary tree in which a decision is made at each internal vertex.

EXERCISE

1. Show that a Hamiltonian path is a spanning tree.
2. Let G be a graph with k components, where each component is a tree. If n is the number of vertices and m is the number of edges in G. Prove that $n = m + k$.
3. Show that the complete bipartite graph $K_{r, s}$ is not a tree if $r \geq 2$.
4. Show that the number of vertices in a binary tree is always odd.
5. Prove that the maximum number of vertices in a binary tree of depth d is $2^d - 1$ when $d \geq 1$.
6. Let $T = (V, E)$ be a complete m–ary tree of height h with l leaves. Then prove that $l \leq m^h$ and $h \geq (\log m^l)$.
7. If $G = (V, E)$ is an undirected graph then G is connected iff G has a spanning tree.
8. If every tree $T = (V, E)$ then prove that $|V| = |E| + 1$.
9. If $G = (V, E)$ be a loop–free undirected graph, and $\deg(v) \geq 2 \; \forall \; v \in V$, prove that G contains a cycle.
10. Explain that, a graph with n vertices, $n - 1$ edges, and no circuit is connected.

Suggested Readings

1. A.V.Aho, *et al*., *Data Structures and Algorithmis,* Addison-Wesley, Reading, MA, 1983, pp. 75-196, 230-292.
2. T.H.Carmen *et al*., *Introduction to Algorithms,* McGraw-Hill, New York, 1990.
3. D.I.A. Cohen, *Basic Techniques of Combinatorial Theory,* Wiley, New York, 1978, pp. 228-282.

4. R.L.Kruse, *Data Structure and Program Design,* 2nd ed., Prentice Hall, Englewood Cliffs, NJ, 1987, pp. 318-498, 512-517.

5. J.A.McHugh, *Algorithmic Graph Theory,* Prentice-Hall, Englewood Cliffs, NJ, 1990.

6. S. Sahni, *Concepts in Discrete Mathematics,* 2nd ed., Camelot, Fridley, MN, 1985, pp. 391-448.

7. S.S. Skieno, *Implementing Discrete Mathematics*, Addison-Wesley, Reading, MA, 1990.

8. S.K. Yadav, *Elements of Graph Theory*, Ane Books New Delhi, 2011.

❑❑❑

12

Planar Graphs

Kazimierz Kuratowski (1896–1980)

Kazimierz Kuratowski (1896-1980) was a Polish mathematician born in Warsaw. His father was a lawyer. He studied engineering in the university of Glasgow, Scotland and completed it in 1913. Due to First World War in 1914, he could not returns to Glasgow. In 1915, Kuratowski studied mathematics at the university of Warsaw under the guidance of the reputed logician Jan Lukasiewicz and graduated in 1919. He received his Ph.D. in 1921.

Kuratowski became professor of mathematics at Lvov Technical University in 1927. Six years later, he returned to the university of Warsaw and hold academic and administrative both posts where he worked untill 1966. He was editor-in-chief of several research journals and council member of various research and apex bodies of that time. Kuratowski wrote numerous articles for professional journals and contributed to topology, analysis and graph theory.

12.1 Introduction

In Graph Theory, a **planar graph** is a graph which can be embedded in the plane *i.e.*, it can be drawn on the plane in such a manner that its edges intersect only at their endpoints. A planar graph which has already drawn in the plane without edge intersection is called a plane graph or planar embedding of the graph. A plane graph can be defined as a planar graph with a mapping from every node to a point in $2D$ space, and from every edge to a pane curve, such that the extreme points of each curve are the points mapped from its end nodes, and all the curves are disjoint except

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_12

on their extreme points. Plane graphs can be encoded by combinatorial maps.

It can easily seen that a graph that can be drawn on the plane can be drawn on the sphere as well and vice-a-versa.

The equivalence class of topologically equivalent drawing on the sphere is called a planar map. Although a planar graph has an external or unbounded face, none of the faces of a planar map have a particular status.

A generalization of planar graphs are graphs which can be drawn on a surface of a given genus. In this terminology, planar graphs have graph genus *O*, since the plane and the sphere are surfaces of genus *O*. The following are the planar graphs. (Fig. 12.1).

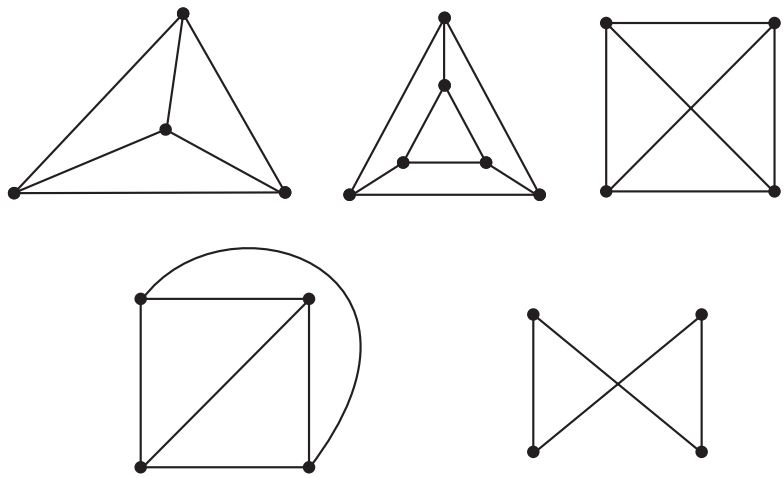


Fig. 12.1: Planar graphs

12.2 Geometrical Representation of Graphs

As we have defined a graph G in chapter 1 as $G = (V, E, \phi)$ where the set V consists of the five objects named a, b, c, d and e *i.e.* $V = \{a, b, c, d, e\}$ and the set E consists of seven objects (none of which is in set V) named 1, 2, 3, 4, 5, 6, and 7, *i.e.*

$$E = \{1, 2, 3, 4, 5, 6, 7\}$$

and the relationship between the two sets is defined by the mapping, Ψ, which consists of combinatorial representation of the graph.

$$\psi = \begin{cases} 1 & \rightarrow (a, c) \\ 2 & \rightarrow (c, d) \\ 3 & \rightarrow (a, d) \\ 4 & \rightarrow (a, b) \\ 5 & \rightarrow (b, d) \\ 6 & \rightarrow (d, e) \\ 7 & \rightarrow (b, e) \end{cases}$$

Where, the symbol $1 \rightarrow (a, c)$ represents that object 1 from set E is mapped onto the pair (a, c) of objects from set V.

It can be represented by means of the Fig. 12.2. It truly shows that the graph can be represented by means of such configuration.

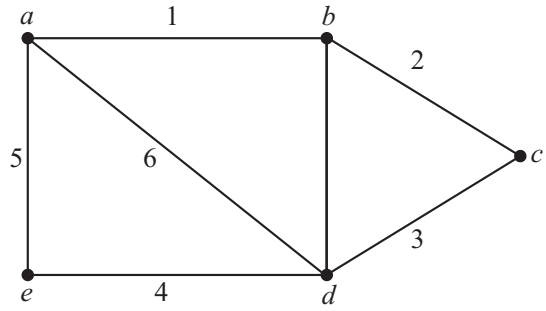


Fig. 12.2: Geometrical Representation of Graph

Theorem 1: The complete graph K_5 is non-planar.

Proof: We attempt to draw K_5 in the plane. We first start with a pentagon: As shown in Fig. 12.3

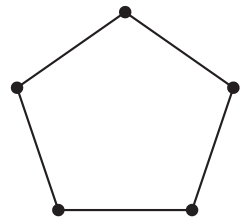


Fig. 12.3 (a)

A complete graph contains an edge between every pair of vertices, so there is an edge between a and c. This may as well be inside the

pentagon (as if it is outside then we just adjust the following argument appropriately).

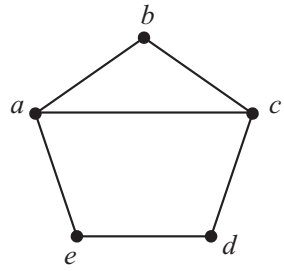


Fig. 12.3 (b)

Now we add the edge between b and e (this must be outside the pentagon as it cannot cross $\{a, c\}$, the edge between a and b (inside so as to not cross $\{b, e\}$), and then between c and e (outside so as to not cross $\{a, d\}$ [As Fig. 12.3(c)]

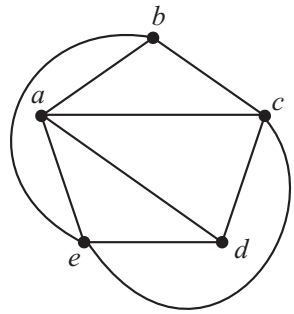


Fig. 12.3 (c)

All these edges were forces into position and we have no choice. It remains to add an edge between b and d. We cannot add it inside (since it would cross $\{a, c\}$ nor can we add it outside (since it would cross $\{c, e\}$).

Consequently K_5 is non-planar. ■

12.3 Bipertite Graph

A simple graph $G = (V, E)$ is called bipertite if $V = V_1 \cup V_2$ with $V_1 \cap V_2 = \phi$ and every edge of G is of the form $\{a, b\}$ with one of the vertices a and b in V_1 and the other in V_2.

If every vertex in V_1 is joined to every vertex in V_2 we can obtain a complete bipertite graph. We may write $K_{m, n}$ for the complete bipertite graph with $|V_1| = m$ and $|V_2| = n$.

Here $|E| = mn$. (see examples in Fig. 12.4)

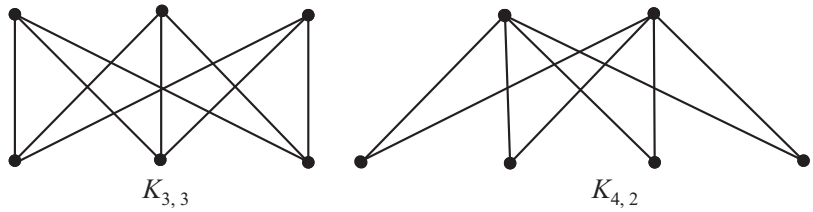


Fig. 12.4: Complete Bipertite Graphs

Theorem 2: The complete bipertite graph $K_{3, 3}$ is nonplanar.

Proof: Let $V_1 = \{a, b, c\}$ and $V_2 = \{x, y, z\}$ and draw a hexagonal circuit:

$a \to x \to b \to y \to c \to z \to a$

The proof is completed by observing that two of the edges $\{a, y\}$, $\{b, z\}$ or $\{c, x\}$ must both lie inside or both outside the hexagon and hence must cross.

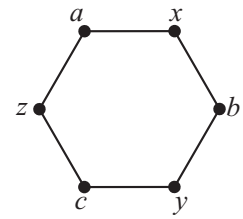


Fig. 12.5

12.4 Homeomorphic Graph

Two graphs G_1 and G_2 such that $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are said to be homeomorphic if and only if G_2 can be obtained from G_1 by the insertion or deletion of a number of vertices of degree two (deg. 2).

The following three graphs are homeomorphic (Fig. 12.6)

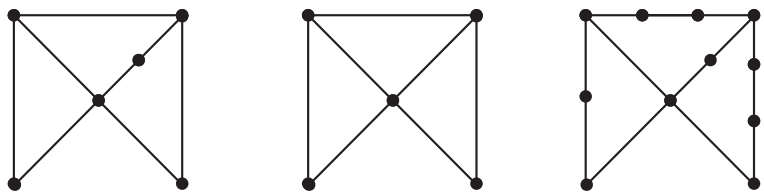


Fig. 12.6: Homeomorphic Graph

One can think of homeomorphic graphs as being as the same shape. Adding or deleting a vertex of degree two does not change the shape of the edges but simply replaces a single edge by a pair of edges taking the same shape (or vice versa).

Example 1

If graph (p_1, q_1) and (p_2, q_2) graph are homeomorphic then $p_1 + q_2 = p_2 + q_1$.

Solution

Let $(p_1, q_1) = G_1$, and $(p_2, q_2) = G_2$ he homeomorphic graphs. Therefore G_1 and G_2 can be from a (p, q) graph G by a series of elementary subdivisions respectively (say r and s).

In each elementary subdivision, the number of prints as well as the number of edges increase by one.

Hence $$p_1 = p + r, q_1 = q + r, p_2 = p + s, q_2 = q + s$$

Hence $$p_1 + q_2 = b + r + q + s = (p + s) + (q + r) = p_2 + q_1$$

(It shows necessary and sufficient condition for a graph to be planar).

■

12.5 Kuratowski's Graphs

The graphs K_5 and $K_{3,3}$ are called Kuratowski's graphs. A graph is planar if and only if it has no subgraph, homeomorphic to K_5 or $K_{3,3}$.

Since planarity is such a fundamental property, it is clearly of importance to know which graphs are planar and which are not.. We have already noted that, in particular K_5 and $K_{3,3}$ are non-planar and that any proper subgraph of either of these graphs is planar. A remarkable simple characterisation of planar graphs was given by Kuratowski (1930). This can be proved by two lemmas.

Lemma 12.1: If G is non-planar, then every subdivision of G is non-planar.

Lemma 12.2: If G is planar, they every subgraph of G is planar.

Proof: Since K_5, and $K_{3,3}$ are nonplanar, we can see that if G is planar, then G cannot contain a subdivision of K_5 or of $K_{3,3,}$ (Fig. 12.7). Kuratowski showed that this necessary condition is also sufficient.

Before proving Kuratowski's theorem, we need to establish two more simple lemmas.

Let G be a graph with a 2-vertex cut $\{u, v\}$. Then there exist edge-disjoint subgraph G_1 and G_2 such that $V(G_1) \cap V(G_2) = \{u, v\}$ and $G_1 \cup G_2 = G$. Consider such a separation of G into subgraphs. In both G_1 and G_2 join u.

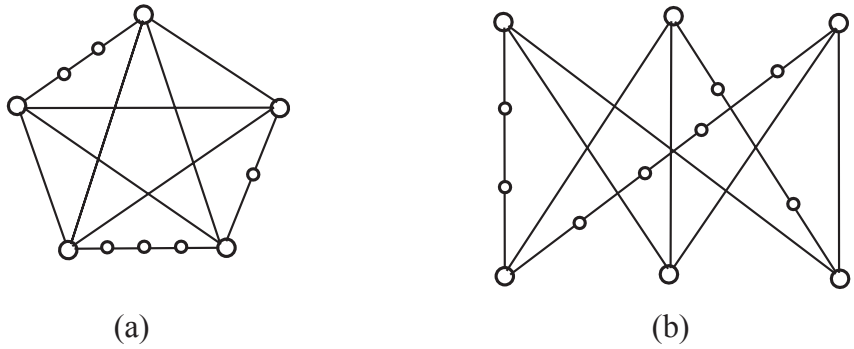


Fig. 12.7: (*a*) A Subdivision of K_5; (*b*) a Subdivision of $K_{3,3}$

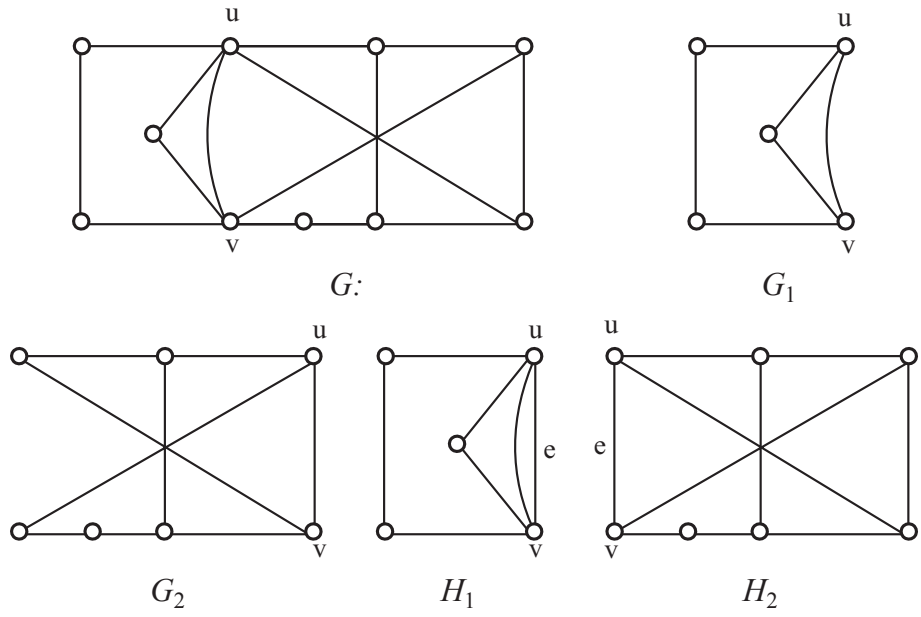


Fig. 12.8

and v by a new edge e to obtain graphs H_1 and H_2, as in Fig. 12.8. Clearly $G = (H_1 \cup H_2) = e$. It is also easily seen that $\varepsilon(H_i) < \varepsilon(G)$ for $i = 1, 2$.

Lemma 3: If G is non-planar, than at least one of H_1 and H_2 is also non-planar.

Proof: Contradictingly, suppose, both H_1 and H_2 are planar Let H_1 be a planar embedding of H_1, and let f be a face of $\bar{H}_1$ incident with e. If $\bar{H}_2$ is an embedding of H_2 in f such that H_1 and $\bar{H}_2$ have only the vertices u and v and the edge e is common, then $\left(\bar{H}_1 \cup H_2\right) - e$ is a planar embedding of G.

This contradiction hypothesis shows that G is nonplanar.

Lemma 4: Let G be a non-planar connected graph that contains no subdivision of K_5 or $K_{3,3}$ and has as few edges as possible. Then G is simple and 3–connected.

Proof: By contradiction. Let G satisfy the hypotheses of the lemma. Then G is clearly a minimal nonplanar graph, and therefore must be a simple block. If G is not 3-connected, let $\{u, v\}$ be a 2-vertex cut of G and let H_1 and H_2 be the graphs obtained from this cut as described above. By lemma at least one of H_1 and H_2, say H_1, is nonplanar. Since $\varepsilon(H_1) < \varepsilon(G)$, H_1 must contain a subgraph K which is a subdivision of K_5 or $K_{3,3}$; moreover $K \not\subseteq G$, and so the edge e is in K. Let P be a (u, v)-path in $H_2 - e$. Then G contains the subgraph $(K \cup P) - e$, which is a subdivision of K and hence a subdivision of K_5 or $K_{3,3}$. This contradiction establishes the lemma. ■

Theorem 3: A graph is planar if and only if it contains no subdivision of K_5 and $K_{3,3}$.

Proof: We have already noted that the necessity follows from lemmas 1 and 2. We shall prove the sufficiency by contradiction.

If possible, choose a nonplanar graph G that contains no subdivision of K_5 or $K_{3,3}$ and has as few edges as possible. From lemma 4 it follows that G is simple and 3–connected. Clearly G must also be a minimal nonplanar graph.

Let $u\ v$ be an edge of G, and let H be a planar embedding of the planar graph $G - u\ v$. Since G is 3-connected, H is 2-connected u and v are contained together in a cycle of H. Choose a cycle C of H that contains u and v and is such that the number of edges in Int C is as large as possible.

Since H is simple and 2-connected, each bridge of C in H must have at least two vertices of attachment. Now all outer bridges of C must be 2-bridges that overlap uv because, if some outer bridge were a k-bridge for $k \geq 3$ or a 2-bridge that avoided uv, then there would be a cycle C' containing u and v with more edges in its interior than C, contradicting the choice of C. These two cases are illustrated in Fig. 12.9 (with C' indicated by heavy lines).

In fact, all outer bridges of C in H must be single edges. For if a 2-bridge with vertices of attachment x and y had a third vertex, the set $\{x, y\}$ would be a 2-vertex cut of G, contradicting the fact that G is 3-connected.

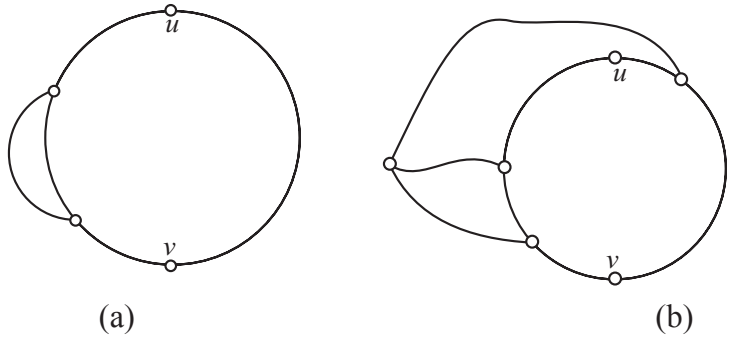


Fig. 12.9: Vertex Attachment

Two cases now arise, depending on whether B has a vertex of attachment different from u, v, x and y or not.

Case 1: B has a vertex of attachment different from u, v, x and y. We can choose the notation so that B has a vertex of attachment v_1 in $C(x, u)$ (see Fig. 12.10). We consider two subcases, depending on whether B has a vertex of attachment in $C(y, v)$ or not.

Case 1(*a*): B has a vertex of attachment v_2 in $C(y, v)$. In this case there is a (v_1, v_2)-path P in B that is internally-disjoint from C. But then $(C \cup P) + \{uv, xy\}$ is a subdivision of $K_{3,3}$ in G, a contradiction (see Fig. 12.10).

Case 1(*b*): B has no vertex of attachment in $C(y, v)$. Since B is skew to uv and to xy, B must have vertices of attachment v_2 in $C\{u, y\}$ and v_3 in $C\{v, x\}$. Thus B has three vertices of attachment v_1, v_2 and v_3. There exists a vertex v_0 in $V(B) \setminus V(C)$ and three paths P_1, P_2 and P_3 in B joining v_0 to v_1, v_2 and v_3, respectively, such that, for $i \neq j$, P_i and P_j have only the vertex v_0 in common. But now $(C \cup P_1 \cup P_2 \cup P_3) + \{uv, xy\}$ contains a subdivision of $K_{3,3}$ a contradiction. This case is illustrated in Fig. 12.11. The subdivision of $K_{3,3}$ is indicated by heavy lines.

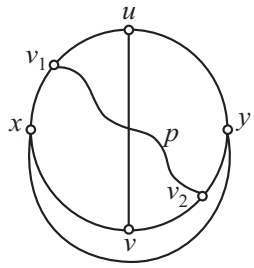


Fig. 12.10

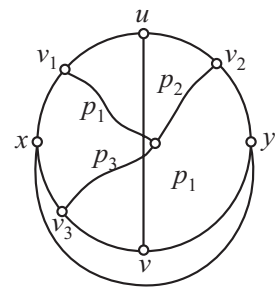


Fig. 12.11

Case 2: B has no vertex of attachment other than u, v, x and y. Since B is skew to both uv and xy, it follows that u, v, x and y must all be vertices of attachment of B. Therefore there exists a (u, v)-path P and an (x, y)-path Q in B such that (*i*) P and Q are internally-disjoint from C, and (*ii*) $|V(P) \cap V(Q)| \geq 1$. We consider two subcase, depending on whether P and Q have one or more vertices in common.

Case 2(*a*): $|V(P) \cap V(Q)| = 1$. In this case $(C \cup P \cup Q) + \{uv, xy\}$ is a subdivision of K_5 in G, again a contradiction (see Fig. 12.12).

Case 2(*b*): $|V(P) \cap V(Q)| \geq 2$. Let u' and v' be the first and last vertices of P on Q, and let P_1 and P_2 denote the (u, u')- and (v', v)-sections of P. Then $(C \cup P_1 \cup P_2 \cup Q) + \{uv, xy\}$ contains a subdivision of $K_{3,3}$ in G, once more a contradiction (see Fig. 12.13).

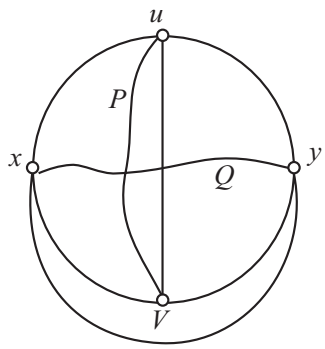


Fig. 12.12

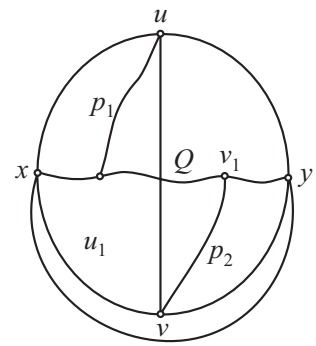


Fig. 12.13

Thus all the possible cases lead to contradictions, and the proof is complete. ■

12.6 Dual Graphs

A plane graph G partition the rest of the plane into a number of connected regions; the closures of these regions are called the faces of G. Fig. 12.14 shows a plane graph with six faces f_1, f_2, f_3, f_4, f_5 and f_6. The notion of a face applies also to embeddings of other surfaces. We denote $F(G)$ and $\phi(G)$ respectively as the set of faces and the number of faces of a plane graph G. Each plane graph has exactly one unbounded face, called the exterior face. In the plane graph of Fig. 12.14, f_1 in the exterior face.

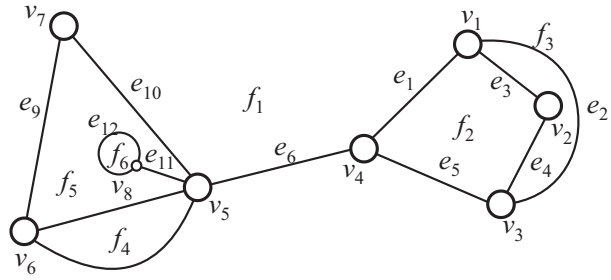


Fig. 12.14: A Plane Graph with six Faces

We denote the boundary of a face f of a plane graph G by $b(f)$. If G is connected, then $b(f)$ can be regarded as a closed walk in which each cut edge of G in $b(f)$ is traversed twice; when $b(f)$ contains no cut edges, it is a cycle of G. For example, in the plane graph of Fig. 12.14

$$b(f_2) = v_1e_3v_2e_4v_3e_5v_4e_1v_1$$

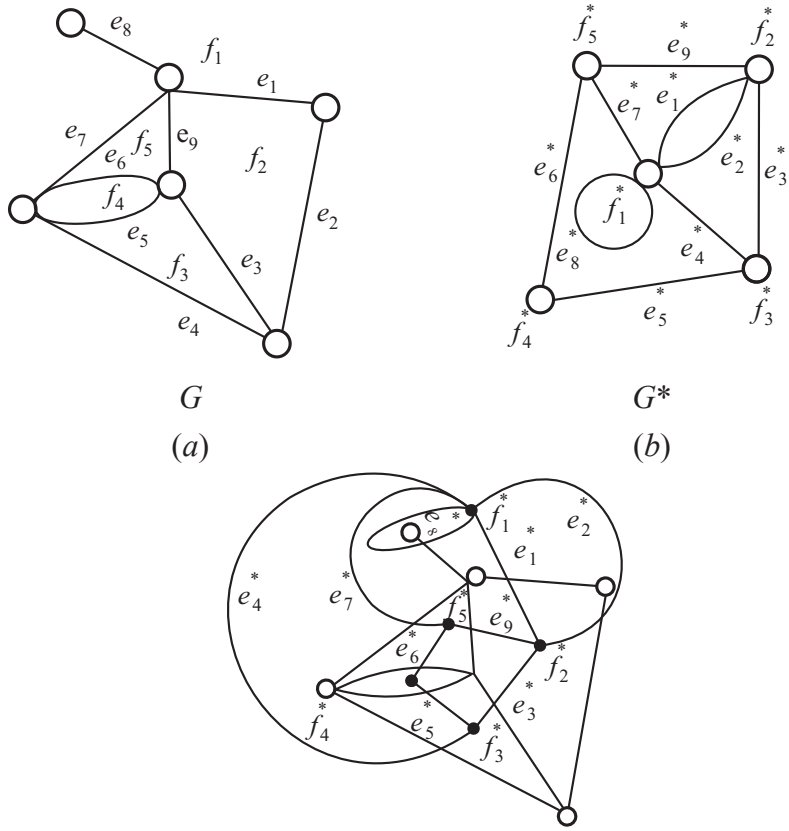
and

$$b(f_5) = v_7e_{10}v_5e_{11}v_8e_{12}v_8e_{11}v_5e_8v_6e_9v_7$$

A face f is said to be incident with the vertices and edges in its boundary. If e is a cut edge in a plane graph, just one face is incident with e; otherwise, there are two faces incident with e. We say that an edge separates the faces incident with it. The degree, $d_G(f)$, of a face f is the number of edges with which it is incident (that is, the number of edges in $b(f)$), cut edges being counted twice. In Fig. 12.14 f_1 is incident with the vertices v_1, v_3, v_4, v_5, v_6, v_7 and the edges e_1, e_2, e_5, e_6, e_7, e_9, e_{10}; e_1 separates f_1 from f_2 and e_{11} separates f_5 from f_5; $d(f_2) = 4$ and $d(f_5) = 6$.

Given a plane graph G, one can define another graph G^* as follows: corresponding to each face f of G there is a vertex f^* of G^*, and corresponding to each edge e of G there is an edge e^* of G^*, two vertices f^* and g^* are joined by the edge e^* in G^* if and only if their corresponding faces f and g are separated by the edge e in G. The graph G^* is called the dual of G.

It is easy to see that the dual G^* of a plane graph G is planar; in fact, there is a natural way to embed G^* in the plane. We place each vertex f^* in the corresponding face f of G, and then draw each edge e^* in such a way that it crosses the corresponding edge e of G exactly once (and crosses no other edge of G). It is intuitively clear that we can always draw the dual as a plane graph in this way, but we shall not prove this fact. Note that if e is a loop of G, then e^* is a cut edge of G^*, and vice versa.



G (*a*) *G** (*b*)

Fig. 12.15: A Plane Graph and its Dual

Note:

1. G^* of a plane graph G as a plane graph we can consider the dual G^{**} of G^* and $G^{**} \equiv G$.
2. Isomorphic plane graphs may have nonisomorphic duals.
3. The following relations are direct consequences of the definition of G^*

   (*i*) $\nu(G^*) = \phi(G)$ (*ii*) $\in(G^*) = \in(G)$

   (*iii*) $d_G^*(f^*) = d_G(f) \quad \forall f \in F(G)$.

12.7 Euler's Formula

Euler's formula states that it a finite, connected, planar graph is drawn in a plane without any edge intersection. If v is the number of vertices, e the number of edges and f, the number of faces (regions bounded by edges, including the outer, infinitely-large region), then

$$v - e + f = 2.$$

In a finite, connected, simple, planar graph, any face (except possibly the outer one) is bounded by at least three edges and every edge touches atmost two faces; using Euler's formula, we can show that these graphs are sparse in the sense that $e \leq 3v - 6$ if $v \geq 3$.

A simple graph is called maximal planar if it is planar but adding any edge would destroy that property. All the faces (even the outer one) are then bounded by three edges, explaining the alternative term triangular for these graphs. If a triangular graph has v vertices with $v > 2$, then it has precisely $3v - 6$ edges and $2v - 4$ faces.

Euler's formula is also valid for simple polyhedra. This is no coincidence, every simple polyhedron can be turn onto a connected, simple, planar graph by using the polyhedroin's edges as edges of the graph. The faces of the resulting planar graph corresponds to the faces of the polyhedron (Fig. 12.16).

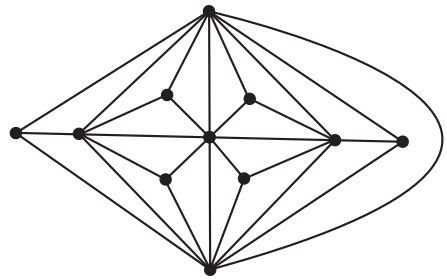


Fig. 12.16: The Goldner–Harary Graph is Maximal Planer all its Faces are Bounded by three Edges

Theorem 4: If G is a connected planar graph then $v - e + f = 2$.

Proof: By induction we can prove on f, the number of faces in G. If $f = 1$, then each edge of G is a cut edge and so G, being connected, is a tree. In this case $e = v - 1$. Suppose that it is true for all connected planar graphs with fewer than n faces, and let G be a connected planar graph with $n \geq 2$ faces. We choose an edge e_1 of G that is not a cut edge.

Then $G - e_1$ is a connected planar graph and has $n - 1$ faces, since the two faces of G separated by e_1 combine to form one face of $G - e_1$. From induction hypothesis, we have

$$v(G - e_1) - e(G - e_1) + f(G - e_1) = 2.$$

and, using the relations

$$v(G - e_1) = v(G)$$
$$e(G - e_1) = e(G) - 1$$
$$f(G - e_1) = f(G) - 1.$$

We set, finally

$$v(G) - e(G) + f(G) = 2.$$

$$\Rightarrow \qquad v - e + f = 2. \qquad \blacksquare$$

Corollary 1: *All planar embeddings of a given connected planar graph have the same number of faces.*

Proof: Let G and H be two planar embeddings of a given connected planar graph.

Since $G \cong H$,

$$v(G) = v(H) \text{ and } e(G) = e(H)$$

using theorem 4, we have

$$f(G) = e(G) - v(G) = 2$$
$$= e(H) - v(H) + 2 = f(H) \qquad \blacksquare$$

Corollary 2: *If G is a simple planar graph with $v \geq 3$, then $e \leq 3v - 6$.*

Proof: It clearly suffices to prove this for connected graphs.

Let G be a simple connected graph with $v \geq 3$.

Then $d(\phi_1) \geq 3$. for all $\phi_1 \in \phi$.

and $$\sum_{\phi_1 \in \phi} d(\phi_1) \geq 3f \qquad \text{(generating graph)}$$

also $$3e \geq 3f.$$

From theorem 4,

$$v - e + \frac{2e}{3} \geq 2.$$

or $$e \leq 3v - 6 \qquad \blacksquare$$

Corollary 3: *If G is a simple planar graph than $\delta \geq 5$.*

Proof: It is trivial for $v = 1, 2$

If $v \geq 3$, then from corollary 2, we have

$$\delta v \leq \sum_{v \in V} d(v) = 2e \sum 6v - 12.$$

It follows that $\delta \leq 5$. $\blacksquare$

Corollary 4: *K_5 and $K_{3,3}$ are non-planar.*

Proof: If K_5 were planar than by cor. 2, we would have

$$10 = e(K_5) \leq 3v(K_5) - 6 = 9$$

This K_5 must be nonplanar

Suppose that $K_{3,3}$ is planar and let G be a planar embedding of $K_{3,3}$

Since $K_{3,3}$ has no cycles of length less than four, every face of G must have degree at least four.

$$4f \leq \sum_{\phi_1 \in \phi} d(\phi_1) = 2e = 18.$$

i.e. $$f \leq 4.$$

From theorem 4, we have

$$2 = v - e + f \leq 6 - 9 + 4 = 1.$$

Which is absurd. ■

12.8 Outerplanar Graphs

A graph is called outerplanar if it has an embedding in the plane such that the vertices lie on a fixed circle and the edges lie inside the disk of the circle and don't intersect. Equivalently, there is some face that includes every vertex. Every outerplanar graph is planar, but the converse is not true: the second example graph shown above (K_4) is planar but not outerplanar. This is the smallest non-outerplanar graph: a theorem similar to Kuratowski's states that a finite graph is outerplanar if and only if it does not contain a subgraph that is an expansion of K_4 (the full graph on 4 vertices) or of $K_{2,3}$ (five vertices, 2 of which connected to each of the other three for a total of 6 edges).

Properties of Outerplanar Graphs

- All finite or countably infinite trees are outerplanar and hence planar.
- An outerplanar graph without loops (edges with coinciding endvertices) has a vertex of degree at most 2.
- All loopless outerplanar graphs are 3-colorable; this face features prominently in the simplified proof of Chvatal's art gallery theorem by *Fisk* (1978). A 3-coloring may be found easily by removing a degree-2 vertex, coloring the remaining graph recursively, and adding back the removed vertex with a color different from its two neighbors.

12.8.1 *k-outerplanar Graphs*

A 1-outerplanar embedding of a graph is the same as an outerplanar embedding. For $k > 1$ a planar embedding is k-outerplanar if removing the vertices on the outer face results in a (k – outerplanar embedding. A graph is k-outerplanar if it has a k-outerplanar embedding.

SOLVED EXAMPLES

Example 1

A connected planar graph has 10 vertices each of degree 3. Into how many regions, does a representation of this planar graph split the plane?

Solution

Given that $n = 10$ and deg of each vertex = 3.

$\Rightarrow \quad \sum \deg(v) = 3 \times 10 = 30.$

But, $\quad \sum \deg(v) = 2e$

$\Rightarrow \quad 30 = 2e \Rightarrow e = 15$

Using Euler's formula, we have

$$n - e + f = 2$$

$\Rightarrow \quad 10 - 15 + f = 2$

$\Rightarrow \quad f = 7$

$\therefore \quad$ no of regions = 7. ■

Example 2

Prove that K_4 and $K_{2,2}$ are planar

Solution

We have in K_4

$$v = 4 \text{ and } e = 6$$

It is obvious that, $6 \leq 3*4 - 6 = 6$.

Hence relation is satisfied for K_4

For $K_{2,2}$, we have $v = 4$ and $e = 4$.

In this case, the related $e \leq 3v - 6$. that is, $4 \leq 3*4 - 6 = 6$ is satisfied.

Hence K_4 and $K_{2,2}$ are planar graphs. ■

Example 3

If every region of a simple planar graph with n-vertices and e-edges embedded in a plane is bounded by K-edges then prove that $e = \dfrac{K(n-2)}{K-2}$

Solution

We know that every region is bounded by K-edges, then f regions are bounded by Kf-edges.

Each edge is counted twice, once for two of its adjacent regions.

We have $2e = Kf$

$$\Rightarrow \quad f = \frac{2e}{K} \qquad \ldots(i)$$

That is, if G is a connected planar graph with n-vertices, e-edges and f-regions then

$$n - e + f = 2.$$

We have from (i)

$$n - e + \frac{2e}{K} = 2$$

$$\Rightarrow \quad nK - eK + 2e = 2K$$

$$\Rightarrow \quad nK - 2K = eK - 2e$$

$$\Rightarrow \quad K(n-2) = e(K-2)$$

$$\Rightarrow \quad e = \frac{K(n-2)}{K-2}$$ ■

Example 4

If G is a graph with 1000 vertices and 3000 edges. What can you concluded about G is planar?

Solution

We know that, a graph is said to be planar if and only if it satisfies the inequality $q \leq 3p - 6$.

Here, we have,

$$p = 1000 \text{ and } q = 3000$$

Then, $3000 \leq 3p - 6.$

i.e. $3000 \leq 3000 - 6.$

$\Rightarrow$ $3000 \leq 2994$

which is not possible.

Hence we can conclude that G is not a planar graph. ■

Example 5

Find a graph G with degree sequence (4, 4, 3, 3, 3, 3) such that

(*i*) *G is planar* (*ii*) *G is not planar.*

Solution

(*i*) We can draw a planar graph with six vertices (with degree sequence 4, 4, 3, 3, 3, 3.) as shown in Fig 12.17.

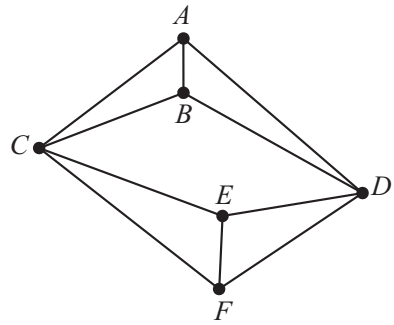


Fig. 12.17

(*ii*) Using Handshaking lamma, we have

$$\sum_{i=1}^{n} \deg(v_i) = 2q$$

$\Rightarrow$ $2q = 4 + 4 + 3 + 3 + 3 + 3$

$\Rightarrow$ $2q = 20$

$\Rightarrow$ $q = 10$

Hence the graph with $p = 6$, is planar iff it satisfy the inequality

$$q = 3p - 6$$

i.e. $10 \leq 3 \times 6 - 6$

$\Rightarrow$ $10 \leq 18 - 6$

$\Rightarrow$ $10 \leq 12$. (which is not possible)

Hence it is not possible to draw a non-planar graph with given degree sequence 4, 4, 3, 3, 3, 3. ■

Example 6

Show that every simple connected planar graph G with less than 12 vertices must have a vertex of degree ≤ 4.

Solution

Let every vertex of G has degree greater than or equal to 5

Then, if $d_1, d_2, d_3,...,d_n$ are the degree of n vertices of G.

We have $d_1 \geq 5, d_2 \geq 5, d_3 \geq 5,...,d_n \geq 5$

So that $d_1 + d_2 + d_3 + ... + d_n \geq 5n$

Or $2m \geq 5n$ (Handshaking property)

Or $$\frac{5n}{2} \leq m \qquad ...(A)$$

On the other hand, we have, $m \leq 3n - 6$.

from (A), we have

$$\frac{5n}{2} \leq 3n - 6 \text{ or } n \geq 12$$

If every vertex of G has degree ≥ 5, then G must have at least 12 vertices

Hence, if G has less than 12 vertices, it must have a vertex of degree < 5. ■

Example 7

What is the minimum number of vertices necessary for a simple connected graph with 11 edges to be planar?

Solution

For a simple connected planar graph $G(m, n)$

we have, $m \leq 3n - 6$.

$$n \geq \frac{1}{3}(m+6)$$

when $m = 11$ we get $n \geq \frac{17}{3}$

Hence the required minimum number of vertices 6. ■

Example 8

Let G be a planar connected graph with n-vertices, m edges and f regions and let its geometice dual G have n* vertices, m* edges and f* regions, then prove that n* = f, m* = m and f* = n.*

Further, it ρ *and* ρ^* *are the ranks and* μ *and* μ^* *are the nullities of G and G* respectively, the prove that* $\rho^* = \mu$ *and* $\mu^* = \rho$.

Solution

Every region of G yields exactly one vertex of G^* and G^* has no other vertex.

The number of regions in G is precisely equal to the number of vertices of G^*

i.e. $$f = n^* \qquad ...(i)$$

Corresponding to every edge e of G, there is exactly one edge e^* of G^* that crosses e exactly once, and G^* has no other edge.

Thus G and G^* have the same number of edges.

i.e. $$m = m^* \qquad ...(ii)$$

Using Euler's formula to G^* and G, we get

$$f^* = m^* - n - 2.$$
$$= m - f + 2.$$
$$= n.$$

Since G and G^* are connected, we get

$$\rho = n - 1,\ \mu = m - n + 1$$
$$\rho^* = n^* - 1,\ \mu^* = m^* - n^* + 1.$$

These together with result (*i*) & (*ii*), and from Euler's formula

$$r^* = n^* - 1 = f - 1 = (m - n + 2) - 1.$$
$$= m - n + 1 = \mu.$$

$$\mu^* = m^* - n^* + 1 = m - f + 1$$
$$= m - (m - n + 2) + 1 = n - 1.$$
$$= \rho.$$ ■

Example 9

If G is a simple connected planar graph s.f. G(p, q) having at least K edges in a boundary of each region. Then prove that (K – 2) q ≤ K (p – 2)

Solution

Every edge on the boundary of G_1 lies in the boundaries of exactly two regions of *G*.

G may have some pendent edges which do not lie in a boundary of any region of *G*.

Sum of lengths of all boundaries of *G* is less than twice the number of edges of *G*.

i.e. $$k f \le 2q \quad ...(i)$$

G is a connected graph, by Euler's formula, we have

$$f = 2 + q - p \quad ...(ii)$$

(*ii*) – (*i*), we set

$$k(2 + q - p) \le 2q$$
$$\Rightarrow \quad (K - 2)q \le K(p - 2)$$ ■

Example 10

Let G be a connected simple planar (n, m) graph is which every region is bounded by at least k edges. Show that $m \le \dfrac{k(n-2)}{(k-2)}$

Solution

Every region in *G* is bounded by at least *k* edges.

We have $$2m \ge k f \quad \ldots(i)$$

Where *f* is the number of regions.

Substituting for f from Euler's formula,

we get,

$$2m \geq k\,(m - n + 2)$$

$$\Rightarrow \qquad K(n-2) \geq km - 2m.$$

$$m \leq \frac{k(n-2)}{(k-2)}$$

■

SUMMARY

1. A path in a connected graph is **Eulerian** if it contains every edge exactly once.
2. A circuit in a connected graph is **Eulerian** if it contains every edge exactly once.
3. A connected graph **Eulerian** if it contains an Eulerian circuit.
4. A connected graph is Eulerian if and only if every vertex of the graph has even degree.
5. A connected graph contains an Eulerian path if and only if it has exactly two vertices of odd degree.
6. A graph is **planar** if it can be drawn in the plane, so not two edges cross.
7. $K_{3,3}$ and K_5 are nonplanar graphs.
8. **Euler's formula** If a connected planar graph with e edges and v vertices partitions the plane into r regions, then $r = e - v + 2$.
9. **Kuratowski's theorem** A graph is planar if and only if it does not contain a subgraph homeomorphic to K_5 or $K_{3,3}$

EXERCISE

1. Prove that a graph which contains a triangle can not be bipertite.
2. Show that C_6 is a bipertite graph.
3. Prove that the sum of the degrees of all regions in a map in equal to twice the number of edges in the corresponding path.
4. Draw all planar graphs with five vertices, which are not isomorphic to each other.

5. Show that of a planar graph G of order n and size m has f regions and K components, then prove that $n - m = f = k + 1$ **[Agra 2003]**

6. Let G be a simple connected graph with fewer than 12 regions, in which each vertex has degree at least 3. Prove that G has a region bounded by at most four edges. **[D.U. 2006]**

7. Show that the condition $m \leq 3n - 6$ is not a sufficient condition for a connected simple graph with n vertices and m edges to be planar **[D.U. 2009]**

8. What is the maximum number of edges possible in a simple connected planar graph with eight vertices? **[Ans.** 18**] [D.U. 2002]**

9. Show that if G is a plane triangulation, then $e = 3v - 6$.

10. Let $S = \{x_1, x_2,...,x_n)$ be a set of $n \geq 3$ points in the plane such that the distance between any two points is at least one. Show that there are at most $3n - 6$ pairs of points at distance exactly one.

Suggested Readings

1. K. Apple and W. Haken, "Every Planar Map is 4-Colorable," *Bulletin of the American Mathematical Society*, Vol. 82 (1976), pp. 711-712.

2. J.A. Bondy and U.S.R. Murty, *Graph Theory with Applications,* Elsevier, New York, 1976.

3. R.A. Brualdi, *Introductory Combinatorics*, 3rd ed., Prentice-Hall, Upper Saddle River, NJ, 1999.

4. A. Tucker, *Applied Combinatorics*, 2nd ed., Wiley, New York, 1984, pp. 3-79, 389-410.

5. D.West, *Introduction to Graph Theory,* 2nd ed., Prentice-Hall, Upper Saddle River, NJ, 2001.

6. R.J.Wilson and J.J. Watkins, *Graphs*: *An Introductory Approach,* Wiley, New York, 1990.

❑❑❑

13

Directed Graphs

Edsger Wybe Dijkstra (1930-2002)

Edsger Wybe Dijkstra (1930-2002) was born in Rotterdam. The Netherlands. He was a computer scientist and mathematician. He graduated from the university of Leyden in 1951 and subsiquently received his Ph.D. in 1956 from Leyden. In 1959 he was awarded by an honorary D.Sc. from Queen´s University, Belfast.

Dijkstra taught mathematics at the Technical university, Nuenen, until 1973 and for next 11 years he had been a research fellow at Burroughs corporation, Nuenen. On 1984, he became the schlumberger centennial chair in computer science at the university of Texas, Austin.

Dijkstra was a distinguished fellow of British computer society. He made outstanding contributions to operating system, PL and Graph theory.

13.1 Introduction

There are so many problems lend themselves naturally to a graph-theoretic formulation. The concept of a graph is not quite adequate. While dealing with problems of traffic flow, it is necessary to know which roads in the network are one-way, and, in which direction the traffic is permitted. It is clear that, a graph of the network is not of much use in such a situation. We need in a graph, in which, each link has an assigned orientation - a *directed graph*.

A directed graph D is an ordered triple (V(D), A(D), ψ_D) consisting of a nonempty set V(D) of vertices, a set A(D), disjoint from V(D), of arcs, and an incidence function ψ_D which associates with each arc of D an ordered pair of vertices of D.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_13

If a is an arc and u and v are vertices such that $\psi_D(a) = (u, v)$, then a is said to join u to v; u is the tail of a, and v is its head. We abbreviate "directed graph" to "digraph".

A digraph D' is a subgraph of D if $V(D') \subseteq V(D)$, $A(D') \subseteq A(D)$ and ψ_D is the restriction of ψ_D to $A(D')$.

Each digraph D can be associated to a graph G on the same vertex set; corresponding to each arc of D there is an edge of G with the same ends. This graph is called the underlying graph of D. As converse, given any graph G, we can obtain a digraph from G by specifying, for each link, an order on its ends. This type of digraph is called an orientation of G.

Digraphs have a simple pictorial representation like graphs. A digraph is represented by a diagram of its underlying graph together with arrows on its edges, each arrow pointing towards the head of the corresponding arc (as shown in Fig. 13.1).

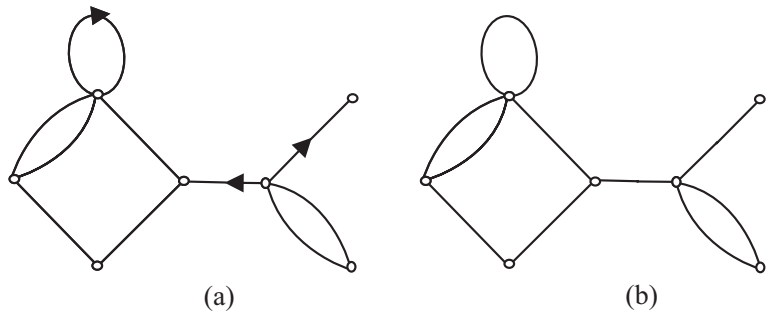


Fig. 13.1: (a) A Digraph ***D***; (b) the Underlying graph ***D***

All the concepts that are valid for graphs automatically applicable to digraphs too. However, there are many concepts that involve the notion of orientation, and these are applicable only to digraphs.

A directed 'walk' in D is a finite non-null sequence $W = (v_0, a_1, v_1,..., a_k, v_k)$, whose terms are alternately vertices and arcs, such that, for $i = 1, 2,...,k$ the arc a_1 has head v_1 and tail v_{i-1}. As with walks in graphs, a directed walk $(v_0, a_1, v_1,...,a_k, v_k)$ is represented simply by its vertex sequence $(v_0, v_1,...,v_k)$.

A directed trail is a directed walk that is a trail; similarly, directed paths, directed cycles and directed tours are defined.

If there is a directed (u, v)-path in D, vertex v is said to be reachable from vertex u in D. Two vertices are diconnected in D if each is reachable

from the other. As in the case of connection in graphs, diconnection is an equivalence relation on the vertex set of D. The subdigraphs $D[V_1]$, $D[V_2]$,...,$D[V_m]$ induced by the resulting partition (V_1, V_2,...,V_m) of $V(D)$ are called the dicomponents of D. A digraph D is diconnected if it has exactly one dicomponent. (see Fig. 13.2)

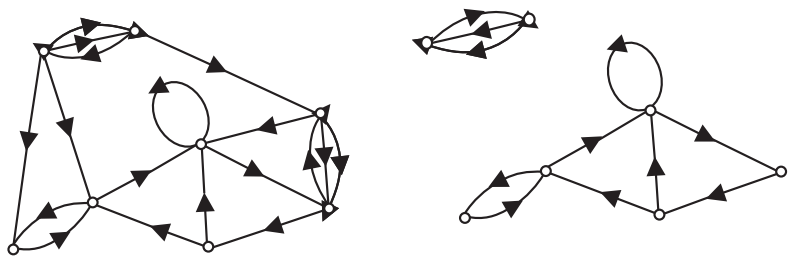


Fig. 13.2: The Digraph of Fig. (a) is not Disconnected and Fig. (b) is has three Dicomponents

The indegree $d_D^-(v)$ of a vertex v in D is the number of arcs with head v; the $d_D^+(v)$ outdegree of v is the number of arcs with tail v. We denote the minimum and maximum indegrees and outdegrees in D by $\delta^-(D)$, $\Delta^-(D)$, $\delta^+(D)$ and $\Delta^+(D)$, respectively. A digraph is strict if it has no loops and no two arcs with the same ends have the same orientation. Throughout this chapter, D will denote a digraph and G its underlying graph. This is a useful convention; it allows us, for example, to denote the vertex set of D by V (since $V = V(G)$), and the numbers of vertices and arcs in D by v and ε, respectively. Also, as with graphs, we shall drop the letter D from our notation whenever possible; thus we write A for $A(D)$, $d^+(v)$ for $d_D^+(v)$ δ^- for $\delta^-(D)$, and so on.

13.2 Directed Paths

In a digraph, There is no close relationship between the lengths of paths and directed paths. We can see from Fig. 13.3, which has no directed path of length greater than one.

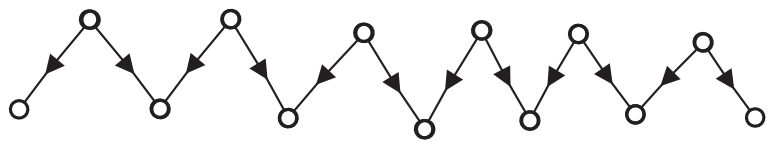


Fig. 13.3

Some information about the length of denoted paths in a digraph can be obtained by looking chromatic numbers.

Theorem 13.1: (Ray and Gallai, 1967-68)

A digraph D contains a directed path of length *X-1***.**

Proof: We consider A' be a minimal set of arcs of D such that $D' = D - A'$ containing no directed cycle and the length of a longest directed path in D' be k. We assign colours 1, 2, 3..., $k + 1$ to the vertices of D' by assigning colour i to vertex v if the length of a longest directed path in D' with origin v is $(i - 1)$.

Denoting by V_i the set of vertices with colour i we can show that $(V_1, V_2,...,V_{k+1})$ is a proper $(k + 1)$ vertex colouring of D.

We can observe that the origin and terminus of any directed path in D' have different colours For Let P be a directed (u, v) - path of positive length in D' and suppose $v \in V_i$. Then, there exists a directed path $Q = (v_1, v_2, v_3...,v_i)$ in D', where $v_i = v$.

Since D' contains no directed cycle, PQ is a directed path with origin u and length at least i,

Thus $u \notin V_i$.

We are to show now, the ends of any arc of D have different colours.

Suppose $(u, v) \in A(D)$.

If $(u, v) \in A(D')$ then (u, v) is a directed path in D' and so u and v have different colours.

Otherwise, $(u, v) \in A'$.

By the minimality of A', $D' + (u, v)$ contains a directed cycle C.

$C - (u, v)$ is a directed (u, v) – path in D' and hence in this case, u and v have different colours.

The above concludes, that $(V_1, V_2, V_3...,V_{k+1})$ is a proper vertex colouring of D.

This follows that

$X \leq k + 1$, and so D has a directed path of length $k \geq X - 1$.

Note: (*i*) The theorem 13.1 is a best possible way to show that every graph G has an orientation in which the longest directed path is of length $X - 1$.

(*ii*) Given a proper X – vertex colouring $(V_1, V_2,..., V_x)$ of G, we orient G by converting edge uv to are (u, v) if $u \in V_i$ and $v \in V_j$ with $i < j$. ■

13.3 Tournament

An orientation of a complete graph is termed as tournament. The tournaments of four vertices are shown in Fig. 13.4. Each of them is to be regarded as indicating the results of games in a round-robin tournament between four players; *e.g.* the first tournament in Fig. 13.4 is shown that one player has won all three games and that the other three have each won one. A directed Hamiltonian path of D is a directed path that includes every vertex of D.

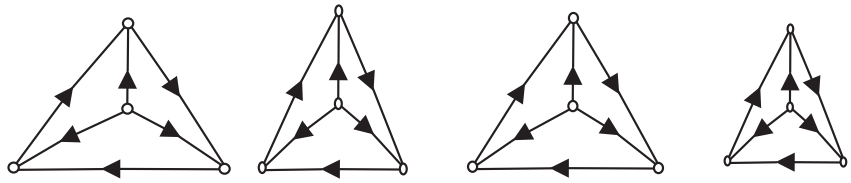


Fig. 13.4: The tournament of four vertices

Theorem 13.2: Every tournament has a directed Hamiltonian path.

Proof: Every tournament has a path such that D is directed. It includes every vertex of D then If D is a tournament, then

$$X = v.$$

Note: (*i*) An interesting fact about tournament is that there is always a vertex from which every other vertex can be reached in at most two steps. An in-neighbour of vertex v in D is a vertex u such that $(u, v) \in A$; and an out neighbour of v is a vertex w such that $(v, w) \in A$.

(*ii*) We denote the sets of in-neighbours and out-neighbours of v in D $N_D^-(v)$ $N_D^+(v)$ by and respectively. ■

Theorem 13.3: A loopless digraph D has an independent set S such that each vertex of D not in S is reachable from a vertex in S by a directed path of length at most two.

Proof: This theorem can be proved by induction on v. The theorem holds trivially for $v = 1$.

We assume that it is true for all digraphs with fewer than v vertices, and let v be an arbitrary vertex of D.

By induction hypothesis there exists in $D' = D - (\{v\} \cup N^+(v))$ an independent set S' such that each vertex of D' not in S' is reachable from a vertex is S' by a directed path of length at most two. If v is an out-neighbour of some vertex u of S', then every vertex of $N^+(v)$ is reachable from u by a directed path of length two.

Hence, in this case, $S = S'$ satisfies the required property.

One the other hand, v is not an out-neighbour of any vertex of S', then v is joined to no vertex of S' and the independent set. $S = S' \cup \{v\}$ has the result.

Note: A tournament contains a vertex from which every other vertex is reachable by a directed path of length at most two, for this, if D is a tournament, then $\alpha = 1$: ■

13.4 Directed Cycles

Theorem 13.2 states that every tournament contains a directed Hamiltonium path. It includes that, if the tournament is assumed to be disconnected.

If S and T are subsets of V, we denote by (S, T) the set of arcs of D that have their tails in S and their heads in T.

Theorem 13.4: (Moon's and Dirac's Theorem) (1966)

Each vertex of a diconnected tournament D with $v \geq 3$ is contained in a directed k – cycle, $3 \leq k \leq v$

Proof: Let D be a disconnected tournament with $v \geq 3$ let u be any vertex of D.

$$\text{Set } S = N^+(u) \text{ and } T = N^-(u).$$

We show that u is in a directed 3 – cycle.

Since D is diconnected, neither S nor T can be empty; and, for the same reason, (S, T) must be non-empty (As shown in fig 13.5). There exists some arc (v, w) in D with $v \in S$ and $w \in T$, and u is in the directed 3 – cycle (u, v, w, u).

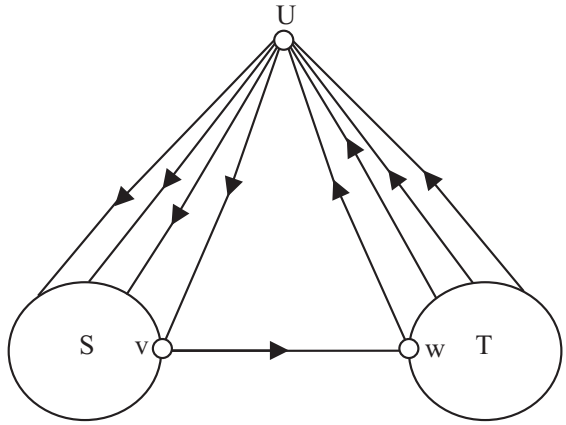


Fig. 13.5: Diconnected Tournament

The theorem can now be proved by induction on k.

Suppose, u is in directed cycle of all lengths between 3 and n, where $n < v$. We can show that u is in directed $(n + 1)$ – cycle.

We consider $C = (v_0, v_1, v_2,...,v_n)$ be a directed n – cycle in which

$$v_0 = v_n = u.$$

If there is a vertex in $V(D)\backslash V(C)$ which is both the head of an arc with tail in C and the tail of an arc with head in C, then there are adjacent vertices v_i and v_{i+1} on C s.t. both (v_i, v) and (v, v_{i+1}) are arcs of D. In this case u is in the directed $(n + 1)$ – cycles. $(v_0, v_1, v_2,...,v_i, v_1, v_{i+1},...,v_n)$.

Otherwise, denoting by S the set of vertices in $V(D)\backslash V(C)$ which are heads of arcs joined to C, and by T the set of vertices in $V(D)\backslash V(C)$ which are tails of arcs joined to C. (as shown in Fig. 13.6).

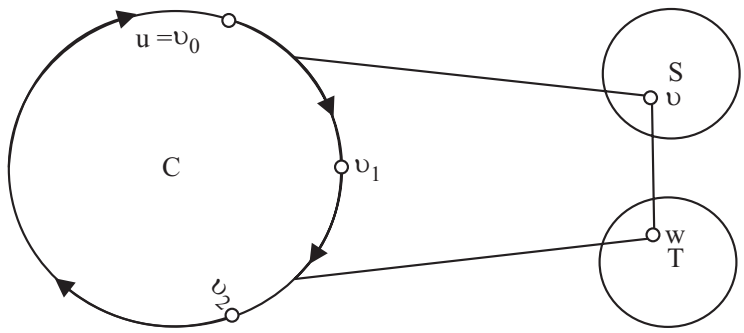


Fig. 13.6

Since D is diconnected, S, T and (S, T) are all nonempty, and there is some arc (v, w) in D with $v \in S$ and $w \in T$.

Hence u is in the directed $(n + 1)$ – cycle $(v_0, v, w, v_2,...,v_n)$.

Note: A directed Hamiltonian cycle of D is a directed cycle that includes every vertex of D.

Theorem 13.5: (Ghouila – Houri's Theorem) (1960)

If D is strict and min $\{\delta^-, \delta^+\} \geq \frac{v}{2} > 1$, then D contains a directed Hamiltonian cycle.

Proof: Let D satisfies the hypotheses of the theorem, but does not contain a directed Hamiltonian cycle.

We denote the length of the longest directed cycle in D by l, and let $C = (v_1, v_2,..., v_l, v_1)$ be a directed cycle in D of length l.

We note that $l > \frac{v}{2}$.

Let P be a longest directed path in $D - V(C)$ and suppose that P has origin u, terminus v and length in (as shown in Fig. 13.7).

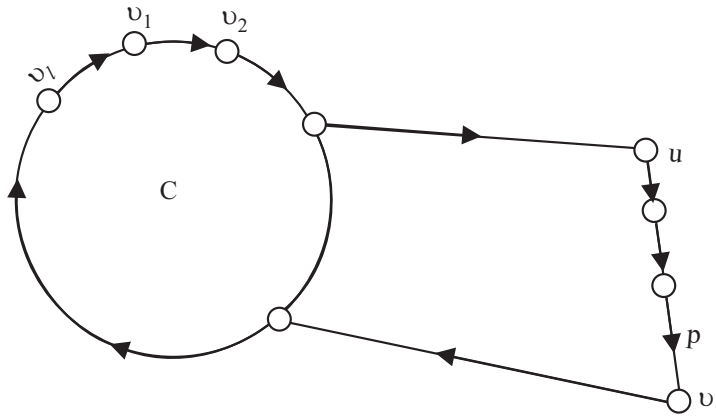


Fig. 13.7

It is clear that,

$$v \geq l + m + 1 \qquad .....(i)$$

Since $l > \frac{v}{2}$

$$\Rightarrow \qquad m < \frac{v}{2} \qquad \ldots(ii)$$

Set $S = \{i \mid (v_{i-1}, u) \in A\}$ and $T = \{i \mid (v, v_i) \in A\}$

We show that S and T are disjoint.

Let $C_{j,k}$ denote the section of C with origin v_j and terminus v_k.

If some integer i were in both S and T, D would contain the directed cycle

$C_{i,\,i-1}\,(v_{i-1}, u)\,P(v, v_i)$ of length $l + m + 1$, contradicting the choice of C

Then, $\qquad S \cap T = \phi \qquad \ldots(iii)$

Now, because P is a maximal directed path in

$$D - V(C),\ N^-(u) \subseteq V(P) \cup V(C).$$

But the number of in - neighbours of u in C is precisely $|S|$ and so

$$d_D^-(u) = d_P^-(u) + |S|$$

since $d_D^-(u) \geq \delta \geq \frac{v}{2}$ and $d_P^-(u) \leq m$,

$$\Rightarrow \qquad |S| \geq \frac{v}{2} - m \qquad \ldots(iv)$$

Similarly,

$$|T| \geq \frac{v}{2} - m \qquad \ldots(v)$$

by (*ii*), both S and T are nonempty.

Adding equ. (*iv*) and (*v*) and using (*i*), we get

$$|S| + |T| \geq l - m + 1.$$

Therefore, by (*iii*)

$$|S \cup T| \geq l - m + 1 \qquad \ldots(vi)$$

Since S and T are disjoint and nonempty, there are positive integers i and k s.t. $i \in S$, $i + k \in T$.

and $\qquad i + j \notin S \cup T$ for $i \subseteq j \subset k \qquad \ldots(vii)$

from (*vi*) & (*vii*), it can be seen that, $k \leq$ m.

Thus the directed cycle

$C_{\,i+k,\,i-1}\,(v_{i+1}, u)\,P\,(v, v_{i+k})$, which has length $l + m + 1 - k$, is longer than C.

Which is a contradiction to establish the theorem. ■

13.5 Acyclic Graph

An orientation D of an undirected graph G is said to be acyclic, if and only if it has no directed cycle.

Let $\alpha(G)$ be the number of acyclic orientation of G.

Theorem 13.6: (Stanley Theorem, 1973)

Let G be a graph of order n. Then the number of the acylic orientations of G is

$$\alpha(G) = (-1)^n \chi_G(-1)$$

Where χ_G is the chromatic polynomial of G.

Proof: This proof is possible by induction on $\in_G$.

Firstly, if G is discrete, then $\chi_G(k) = k^n$, and $\alpha(G) = 1 = (-1)^n$ $(-1)^n = (-1)^n\chi_G(-1)$ as required Now $\chi_G(k)$ is a polynomial that satisfies the recurrence $\chi_G(k) = \chi_G - e(k) - \chi_{G*e}(k)$. To prove the claim, we have to show that $\alpha(G)$ satisfy the same recurrence.

If $$\alpha(G) = \alpha(G - e) + \alpha(G * e) \quad ...(i)$$

Then, by induction hypothesis,

$$\alpha(G) = (-1)^n \chi_{G-e}(-1) + (-1)^{n-1} \chi_{G*e}(-1)$$

$$= (-1)^n \chi_G(-1)$$

for equ. (i) we can see that every acyclic orientation of G gives an acylic orientation of $G - e$. On the other hand, if D is an acyclic orientation of $G - e$ for $e = u\,v$, it extends to an acylic orientation of G by putting e_1: $u \to v$ or e_2: $v \to u$.

If D has no directed path $u \xrightarrow{*} v$, we choose e_2 and if D has no acyclic, it cannot have both ways $u \xrightarrow{*} v$ and $v \xrightarrow{*} u$

We can conclude that

$$\alpha(G) = \alpha(G - e) + b,$$

where b is the number of acyclic orientation D of $G - e$ that extend in both ways e_1 and e_2. The acyclic orientation D that extend in both ways are exactly those that contain.

Neither $u \xrightarrow{*} v$ nor $v \xrightarrow{*} u$ as a directed path ...(ii)

Each acyclic orientation of $G^* e$ corresponds in a natural way to an acyclic orientation D of $G - e$ that satisfies (*ii*).

Therefore $b = a(G * e)$ ■

13.6 Di-Orientable Graph

A Graph G is said to be di-orientable, if and only if there is a di-connected oriented graph D such that $G = U(D)$

Theorem 13.7: (Robbins Theorem, 1939)

A connected graph G is diorientable if and only if G has no bridges.

Proof: If G has a bridge e, then any orientation of G has atleast two dicomponents.

Suppose that G has no bridges, Hence G has a cycle C, and *a* cycle is always di-orientable.

Let $H \subseteq G$ be maximal such that it has a di-oriented D_H. If $H = G$, then it has done.

Otherwise, there exists an edge $e = v^u \in E_G$ such that $u \in H$ but $v \notin H$ (because G is connected).

The edge e is not a bridge and thus there exists a cycle.

$$C' = \text{e}\, PQ : v \to u \xrightarrow{*} w \xrightarrow{*} v$$

in G, where w is the last vertex inside H

We can observe in Fig. 13.8. In the di-orientation D_H of H there is a directed path

$$P' : u \xrightarrow{*} w.$$

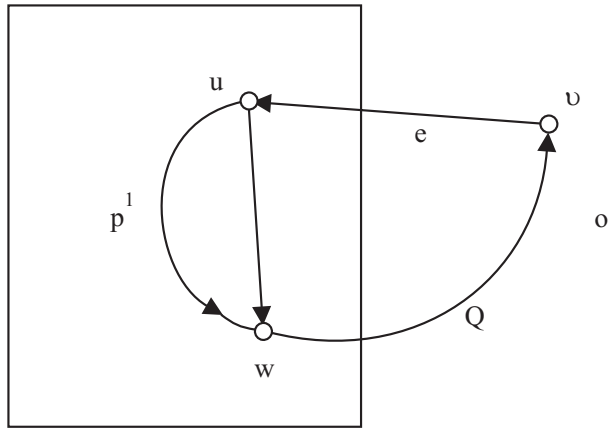


Fig. 13.8

We can orient $e : v \to u$ and the edges of Q in the direction

$$Q : w \xrightarrow{*} v \text{ to obtain a directed}$$

cycle e $P'Q : v \to u \xrightarrow{*} w \xrightarrow{*} v$.

We can conclude that, $G[V_H \cup V_C]$ has a di-orientation, which contradicts the maximality assumption on H. ■

Note: Let D be a directed graph. A directed Euler tour of D is a directed closed walk that uses each edge exactly once. A directed Euler trail of D is a directed walk that uses each edge exactly once.

Example 13.1

Consider a tournament of six teams 1, 2, 3...,6. and let T be the scoring digraph as in the Fig. 13.9. Here 1 → 2 → 4 → 5 → 6 → 3 is a directed Hamilton path, but this extends to a directed Hamilton Cycle (by adding 3 → 1) so, for every team there is a Hamilton path, where it is a winner, and in another, it is a looser?

Solution:

We consider, $s_1(j) = d_T(j)$ be the winning number of the team j (the number of teams beaten by j).

In the above tournament

$$s_1(1) = 4,\ s_1(2) = 3,\ s_1(3) = 3,\ s_1(4) = 2,$$
$$s_1(5) = 2,\ s_1(6) = 1.$$

so, is the team 1 a winner? If so, is 2 or 3 next?

We define the second level scoring for each team by

$$s_2(j) = \sum_{ji \in E_T} s_1(i) \qquad ...(i)$$

This tells us how good team j beat.

In the above figure, we have

$$s_2(1) = 8,\ s_2(2) = 5,\ s_2(3) = 9,$$
$$s_2(4) = 3,\ s_2(5) = 4 \quad \text{and} \quad s_2(6) = 3.$$

Now, it seems that 3 in the winner, but 4 and 6 have the same score. We can continue by defining inductively the m^{th}-level scoring by

$$s_m(j) = \sum_{ji \in E_T} s_{m-1}(i) \qquad ...(ii)$$

It can be proved by matrix method that for a di-connected tournament with atleast four teams, the level scorings will eventually stabilize in a ranking of the tournament; there exists an m for which the mth level scoring gives the same ordering as do the $(m + k)$th-level scorings for all $k \geq 1$.

If T is not di-connected;, then the level scoring should be carried out with respect to the di-components.

In the example the level scoring gives

$$1 \rightarrow 3 \rightarrow 2 \rightarrow 5 \rightarrow 4 \rightarrow 6$$

as the ranking of the tournament. ■

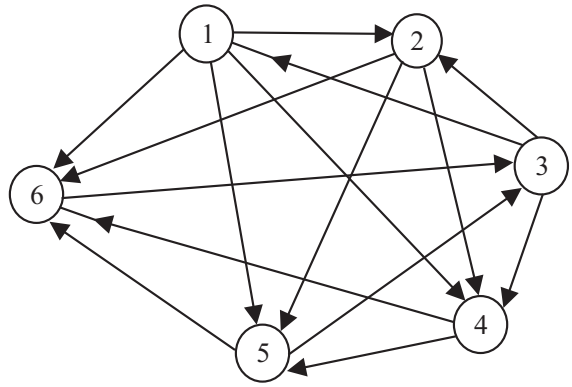


Fig. 13.9

Theorem 13.8: If G is a 2-edge-connected, then G has a di-connected orientation.

Proof: Let G be 2-edge-connected. Then G contains a cycle G_1. We define inductively a sequence $G_1, G_2,...$ of connected subgraphs of G as follows: if G_i $(i = 1, 2,...)$ is not a spanning subgraph of G, let v_i be a vertex of G not in G_i. Then there exist edge-disjoint paths P_i and Q_i from v_i to G_i. Define

$$G_{i+1} = G_i \cup P_i \cup Q_i$$

Since $v(G_{i+1}) > v(G_i)$, this sequence must terminate in a spanning subgraph G_n of G.

We now orient G_n by orienting G_1 as a directed cycle, each path P_i as a directed path with origin v_i, and each path Q_i as a directed path with terminus v_i. Clearly every G_i, and hence in particular G_n, is thereby given a diconnected orientation. Since G_n is a spanning subgraph of G it follows that G, too, has a diconnected orientation. ■

Theorem 13.9: Let G be a 2k-edge-connected graph with an Euler trail. Then G has a k-arc-connected orientation.

Proof: Let $v_0e_1v_1...e_\varepsilon v_\varepsilon$ be an Euler trail of G. Orient G by converting the edge e_i with ends v_{i-1} and v_i to an arc a_i with tail v_{i-1} and head v_i, for $1 \leq i \leq \varepsilon$. Now let $[S, \overline{S}]$ be an m-edge cut of G. The number of times the directed trail $(v_0, a_1, v_1,...,a_\varepsilon, v_\varepsilon)$ crosses from S to $\overline{S}$ differs from the number of times it crosses from $\overline{S}$ to S by at most one. Since it includes all arcs of D, both $(S, \overline{S})$ and $(\overline{S}, S)$ must contain at least $[m/2]$ arcs. ■

Theorem 13.10: Let D be a di-connected tournament with $v \geq 5$, and let A be the adjacency matrix of D. Then $A^{d+3} > 0$, where d is the directed diameter of D.

Proof: The (i, j)th entry of A^k is precisely the number of directed (v_i, v_j)-walks of length k in D. We must therefore show that, for any two vertices v_i and v_j (possibly identical), there is a directed (v_i, v_j)-walk of length $d + 3$.

Let $d_{ij} = d(v_i, v_j)$

Then $0 \leq d_{ij} \leq d \leq v - 1$ and therefore

$3 \leq d - d_{ij} + 3 \leq v + 2$

If $d - d_{ij} + 3 \leq v$ then, there is a directed $(d - d_{ij} + 3)$-cycle C containing v_j. A directed (v_i, v_j)-path P of length d_{ij} followed by the directed cycle C together form a directed (v_i, v_j)-walk of length $d + 3$, as desired.

There are two special cases. If $d - d_{ij} + 3 = v + 1$, then P followed by a directed $(v - 2)$-cycle through v_i followed by a directed 3-cycle through v_i constitute a directed (v_i, v_j)-walk of length $d + 3$ (the $(v - 2)$-cycle exists since $v \geq 5$); and if $d - d_{ij} + 3 = v + 2$, then P followed by a directed $(v - 1)$-cycle through v_i followed by a directed 3-cycle through v_i constitute such a walk. ■

13.7 Applications of Directed Graphs

13.7.1 Job Sequencing Problem

Let a number of jobs $J_1, J_2, J_3..., J_n$ have to be processed on a machine; for example, each J_i might be an order of bottles or jars in a glass factory. After each job the machine must be adjusted to fit the requirements of the next job.

If the time of adaptation from job J_i to job J_i is t_{ij}, find a sequencing of the jobs that minimises the total machine adjustment time.

This problem is clearly related to the travelling salesman problem, and no efficient method for its solution is known. It is therefore desirable to have a method for obtaining a reasonably good (but not necessarily optimal) solution.

Step 1: Construct a digraph D with vertices $v_1, v_2,...,v_n$, such that $(v_i, v_j) \in A$ if and only if $t_{ij} \leq t_{ji}$. By definition, D contains a spanning tournament.

Step 2: Find a directed Hamilton path $(v_{i1}, v_{i2},..., v_{in})$ of D, and sequence the jobs accordingly.

Since step 1 discards the larger half of the adjustment matrix $[t_{ij}]$, it is a reasonable supposition that this method, in general, produces a fairly good job sequence. Note, however, that when the adjustment matrix is symmetric, the method is of no help whatsoever.

As an example, suppose that there are six jobs J_1, J_2, J_3, J_4, J_5 and J_6 and that the adjustment matrix is

	J_1	J_2	J_3	J_4	J_5	J_6
J_1	0	5	3	4	2	1
J_2	1	0	1	2	3	2
J_3	2	5	0	1	2	3
J_4	1	4	4	0	1	2
J_5	1	3	4	5	0	5
J_6	4	4	2	3	1	0

The sequence $J_1 \to J_2 \to J_3 \to J_4 \to J_5 \to J_6$ requires 13 units in adjustment time. To find a better sequence, construct the digraph D as in step 1. (Fig. 13.10)

$(v_1, v_6, v_3, v_4, v_5, v_2)$ is a directed Hamilton path of D, and yields the sequence

$$J_1 \to J_6 \to J_3 \to J_4 \to J_5 \to J_2.$$

which requires only eight units of adjustment time. Note that the reverse sequence

$$J_2 \to J_5 \to J_4 \to J_3 \to J_6 \to J_1$$

is far worse, requiring 19 units of adjustment time.

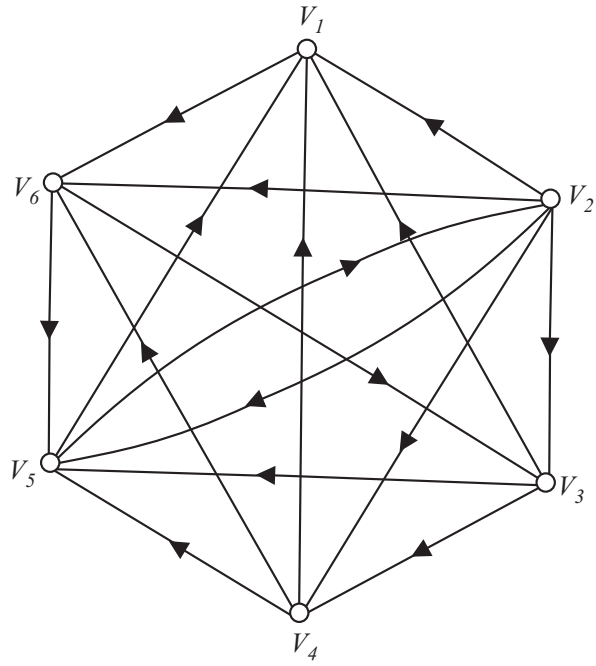


Fig. 13.10: Segnential Digraph

13.7.2 To Design an Efficient Computer Drum

The position of a rotating drum is to be recognised by means of binary signals produced at a number of electrical contacts at the surface of the drum. The surface is divided into 2^n sections, each consisting of either insulating or conducting material. An insulated section gives signal 0 (no current), whereas a conducting section gives signal 1 (current). For example, the position of the drum in Fig.13.11 gives a reading 0010 at the four -contacts.

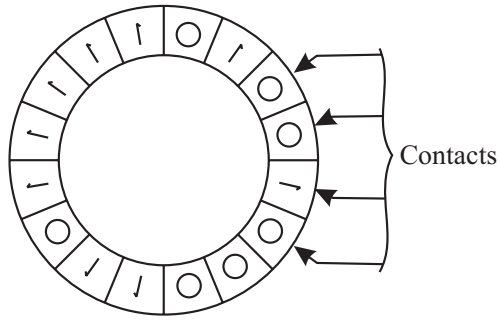
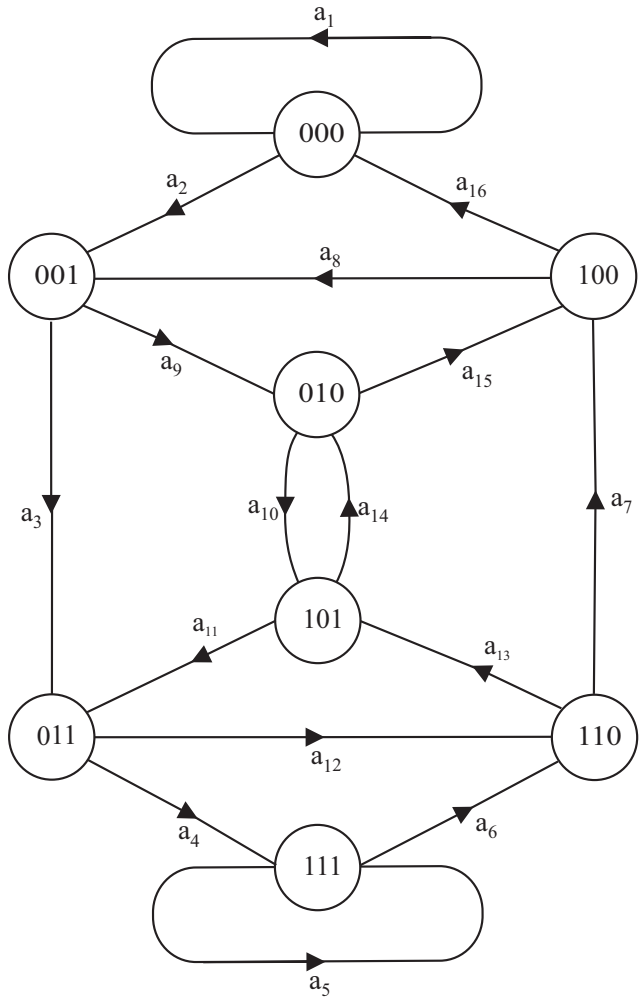


Fig. 13.11: A Computer Drum



Arc	Label
a_1	0000
a_2	0001
a_3	0011
a_4	0111
a_5	1111
a_6	1110
a_7	1100
a_8	1001
a_9	0010
a_{10}	0101
a_{11}	1011
a_{12}	0110
a_{13}	1101
a_{14}	1010
a_{15}	0100
a_{16}	1000

Fig. 13.12: Binary Form

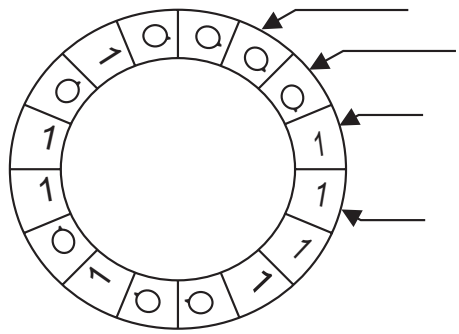


Fig. 13.13: Signential Drum

If the drum were rotated clockwise on section, the reading would be 1001. Thus these two positions can be distinguished, since they give different readings. However, a further rotation of two sections would result in another position with reading 0010, and therefore this latter position is indistinguishable from the initial one.

We design the drum surface in such a way that the 2^n different positions of the drum can be distinguished by k contacts placed consecutively around part of the drum, and we would like this number k to be as small as possible.

First note that k contacts yield a k-digit binary number, and there are 2^k such numbers. Therefore, if all 2^n positions are to give different readings, we must have $2^k \geq 2^n$, that is, $k \geq n$. We shall show that the surface of the drum can be designed in such a way that n contacts suffice to distinguish all 2^n positions.

We define a digraph D_n as follows: the vertices of D_n are the $(n-1)$-digit binary numbers $p_1p_2...p_{n-1}$ with $p_i = 0$ or 1. There is an arc with tail $p_1p_2...p_{n-1}$ and head $q_1q_2...q_{n-1}$ if and only if $p_{i+1} = q_i$ for $1 \leq i \leq n-2$; in other words, all arcs are of the form $(p_1p_2...p_{n-1}, p_2p_3...p_n)$. In addition, each arc $(p_1p_2...p_{n-1}, p_2p_3...p_n)$ of D_n is assigned the label $p_1p_2...p_n$. D_4 is shown in Fig.13.12

Clearly, D_n is connected and each vertex of D_n has indegree two and outdegree two. Therefore, D_n has a directed Euler tour. This directed Euler tour, regarded as a sequence of arcs of D_n, yields a binary sequence of length 2^n suitable for the design of the drum surface.

For example, the digraph D_4 of Fig.13.12 has a directed Euler tour $(a_1, a_2,...,a_{16})$, giving the 16-digit binary sequence 0000111100101101. (Just read off the first digits of the labels of the a_i.) A drum constructed from this sequence is shown in Fig.13.13.

13.7.3 Ranking of the Participants in a Tournament

A number of players each play one another in a tennis tournament. Given the outcomes of the games, how should the participants be ranked?

We consider, for example, the tournament of Fig.13.13. This represents the result of a tournament between six players; we see that player 1 beat players 2, 4, 5 and 6 and lost to player 3, and so on.

One possible approach to ranking the participants would be to find a directed Hamilton path in the tournament, and then rank according to

the position on the path. For instance, the directed Hamilton path (3, 1, 2, 4, 5, 6) would declare player 3 the winner, player 1 runner-up, and so on. This method of ranking, does not bear further examination, since a tournament generally has many directed Hamilton paths; our example has (1, 2, 4, 5, 6, 3), (1, 4, 6, 3, 2, 5) and several others.

Another approach would be to compute the scores (numbers of games won by each player) and compare them. We obtain the score vector as:

$$s_1 = (4, 3, 3, 2, 2, 1)$$

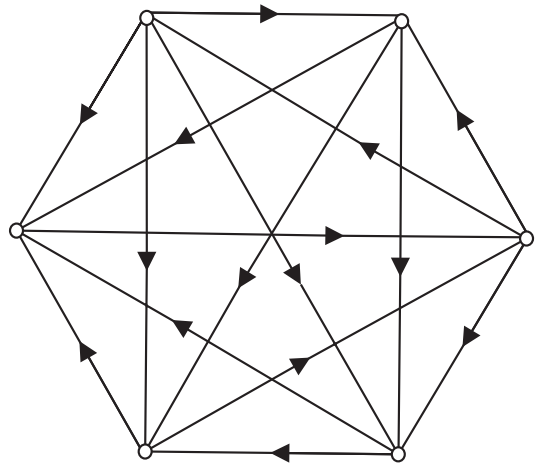


Fig. 13.13: Ranking Problem

This score vector does not distinguish between players 2 and 3 even though player 3 beat players with higher scores than did player 2. We are thus led to the second-level score vector

$$s_2 = (8, 5, 9, 3, 4, 3)$$

in which each player's second-level score is the sum of the scores of the players he beat. Player 3 now ranks first. Continuing this procedure we obtain further vectors as under:

$$s_3 = (15, 10, 16, 7, 12, 9)$$

$$s_4 = (38, 28, 32, 21, 25, 16)$$

$$s_4 = (90, 62, 87, 41, 48, 32)$$

$$s_6 = (183, 121, 193, 80, 119, 87)$$

The ranking of the players is seen to fluctuate a little, player 3 vying with player 1 for first place. This procedure always converges to a fixed

ranking when the tournament in question is disconnected and has at least four vertices. This will lead to a method of ranking the players in any tournament.

Note: In a diconnected digraph D, the length of a shortest directed (u, v)-path is denoted by $d_D(u, v)$ and is called the distance from u to v; the directed diameter of D is the maximum distance from any one vertex of D to any other.

13.8 Network Flows

Various transportation networks or pipelines are conveniently represented by weighted directed graphs. These networks possess some additional requirements. Goods are transported through specific places or warehouses to final locations or market places through a network of roads.

A network N consists of:

(*i*) An underlying digraph $D = (V, E)$

(*ii*) Two distinct vertices *s* and *r*, known as the source and the sink of *N*.

(*iii*) A capacity function $a : V \times V \to IR_+$ for which

$$\alpha(e) = 0, \text{ iff } e \notin E.$$

We denote $V_N = V$ and $E_N = E$.

Let $A \subseteq V_N$ be a set of vertices

and $f : V_N \times V_N \to IR$ any function such that $f(e) = 0$ if $e \notin E_N$.

We can adopt the following notations:

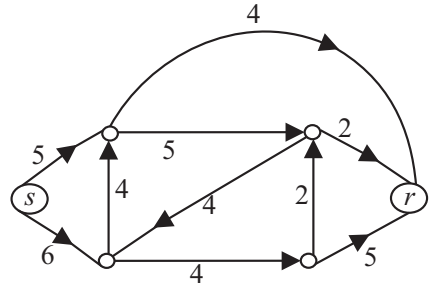


Fig. 13.14

$$[A, \bar{A}] = \{e \in E_D \mid e = uv, u \in A, v \notin A|\}.$$

$$f^+(A) = \sum_{e \in [A, \bar{A}]} f(e) \text{ and } f^-(A)$$

$$= \sum_{e \in [\bar{A}, A]} f(e).$$

Particularly,

$$f^+(u) = \sum_{v \in N} f(uv)$$

and $$f^-(u) = \sum_{v \in N} f(uv)$$

Hence

A flow in a network N is a function

$f: V_N \times V_N \to IR$ such that $o \le f(e) \le a(e) \ \forall \ e$

and $$f^-(v) = f^+(v) \ \forall \ v \notin \{s, r\}$$

Note: (*i*) *Every network N* has a zero flow defined by $f(e) = o \ \forall \ e$.

(*ii*) For a flow of and each subset $A \subseteq V_N$, define the resultant flow from A and the value of f as the number $\text{val}(f_A) = f^+(A) - f^-(A)$ and $\text{val}(f) = val\ (f_s) = f^+(s) - f^-(s)$

(*iii*) A flow f of a network N is a maximum flow, if there does not exist any flow f' such that $val(f) < val(f')$

13.9 Improvable Flows

Let f be a flow in a network N, and let $P = e_1\ e_2\ e_3 ... e_n$ be an undirected path in N where an edge e_i is along p, if $e_i = v_i\ v_{i+1} \in E_N$, and against P, if $e_i = v_{i+1}\ v_i \in E_N$

we define a non-negative number $l(p)$ for P as follows:

$$l(P) = \min e_i\ l(e),$$

where $$l(e) = \{\alpha(e) - f(e)$$

for e is along $f(e)$. for e is againt $P\}$

We may define,

Let f be a flow in a network N. A path $P: s \xrightarrow{*} r$ is (f^-) improvable, if $l(p) > 0$

On the right, the bold path has value $l(P) = 1$ and therefore this path is improvable. (see fig 13.15)

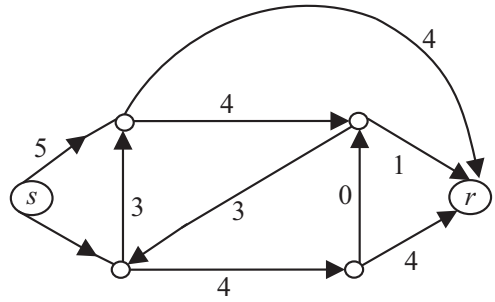


Fig. 13.15

Lemma 13.1: Let $N = (D, s, r, \alpha)$ be a network with a flow f

(*i*) *If* $A \subseteq N \setminus \{s, r\}$, then $\text{val}(f_A) = 0$

(*ii*) $val(f) = -\text{val}(f_r)$.

Proof: Let $A \in N \setminus \{s, r\}$. Then

$$0 = \sum_{v \in A} (f^+(v) - f^-(v))$$

$$= \sum_{v \in A} (f^+(v) - \sum_{v \in A} f-(v)$$

$$= f^+(A) - f^-(A)$$

$$= \text{val}\,(f_A)$$

where the third equality holds since the values of the edge uv with $u, v \in A$ cancel each out.

Similarly we can prove second claim. ■

Lamma 13.2: Let N be a network. If f be a maximum flow of N, then it has no improvable paths.

Proof: We define,

$$f'(e) = \begin{cases} \{f(e) + l(p) \text{ if } e \text{ is along } P \\ f(e) - l(p) \text{ if } e \text{ is against } P \\ f(e) \text{ if } e \text{ is not in } P \end{cases}$$

Than f' is a flow, since at each intermediate vertex $v \notin \{r, s\}$, we have $(f')^-(v) = f(f')^+(v)$. and the capacities of the edges are not exceeded.

Now $\text{val}(f') = \text{val}\,(f) + l(p)$, since p has exactly one edge $sv \in E_N$ for the source s. Hence, if $l(P) > 0$, then we can improve the flow. ■

13.10 Max-Flow Min-Cut Theorem

Let $N = (d, s, r, \alpha)$ be a network. For a subset $S \subset V_N$ with $s \in S$ and $r \notin S$,

let the cut by S be

$$[S] = [S, \overline{S}\,] \; (= \{uv \in E_N \mid u \in S, u \notin S\})$$

The capacity of the cut $[S]$ is the sum

$$\alpha[S] = \alpha^{+}(S) = \sum_{e \in [S]} \alpha(e)$$

A cut $[S]$ is a minimum cut, if there is no cut $[R]$ with $\alpha[R] < \alpha[S]$ for example in Fig. 13.17. The capacity of the cut for the indicate vertices is equal to 10.

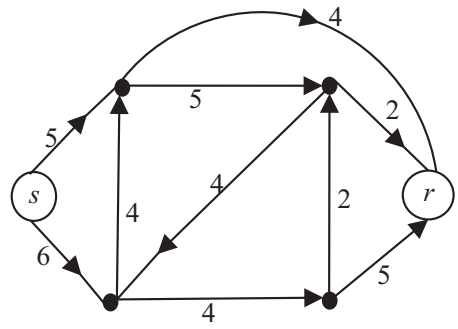


Fig. 13.17

Lemma 13.3: For a flow f and a cut $[S]$ of N,

$$\text{val}(f) = \text{val}\,(f_s) = f^{+}(S) - f^{-}(S)$$

Proof:

Let $S_i = S \setminus \{s\}$

Now, val $(S_i) = 0$ [since $S_i \subseteq N \setminus \{s, r\}$]

and $\text{val}\,(f) = \text{val}\,(f_s)$

$$\text{val}(f_s) = \text{val}(f_s) - \sum_{v \in s_i} f(s,v) + \sum_{v \in s_i} f(v,s)$$

$$+ \text{val}(f_{si}) + \sum_{v \in s_i} f(sv) - \sum_{v \in s_i} f(vs)$$

$$= \text{val}(f_s) = \text{val}(f)$$

■

Theorem 13.11: For a flow f and any cut $[S]$ of N, $\text{val}(f) \leq \alpha[S]$. Furthermore, equality holds if and only if for each $u \in S$ and $v \notin S$,

(*i*) if $e = uv \in E_N$, then $f(e) = a(e)$

(*ii*) if $e = uv \in E_n$, then $f(e) = 0$.

Proof: We have from definition of flow,

$$f^+(S) = \sum_{e \in [s]} f(v) \leq \sum_{e \in [s]} \alpha(e) = a[S] \qquad ...(i)$$

and $f^-(S) \geq 0$

Also $\text{val}(f) = \text{val}(f_S) = f^+(S) - f^-(S)$ [Lemma 13.3]

Hence, $\text{val}(f) \leq \alpha[S]$, as required

Also, the equality $\text{val}(f) = \alpha[S]$ holds iff

(1) $f^+(S) = \alpha[S]$

and (2) $f^-(S) = 0$ ...(*ii*)

This holds iff $f(e) = \alpha(e) \ \forall e \in [S]$ ($\because f(e) \leq \alpha(e)$)

and (2) $f(e) = 0 \ \forall e = uv$

with $u \in S$ and $v \notin S$.

This proves the claim.

Note: If f is a maximum flow and $[S]$ a minimum cut, then $\text{val}(f) \leq \alpha[S]$ ■

13.11 *k*-flow

A k-flow (H, α) of an undirected graph G is an orientation H of G together with an edge colouring $\alpha : E_H \to [0, k\text{-}1]$

such that for all vertices $u \in V$.

$$\sum_{f = uv \in E_H} \alpha(f) = \sum_{e = uv \in E_H} \alpha(e) \qquad ...(i)$$

i.e. The sum of the incoming values equals the sum of outgoing values. A k-flow is nowhere zero, if $\alpha(e) \neq 0 \ \forall e \in E_H$.

In the k-flows we do not have any source or sink for our convenience,

let $\alpha(e^{-1}) = -\alpha(e) \ \forall e \in E_H$

in the orientation H of G so that the condition (*i*) becomes

$$\sum_{e=uv\in E_H} \alpha(e) = 0 \qquad \ldots(ii)$$

The condition (*ii*) generalizes to the subset $A \leq V_G$ in a natural way

$$\sum_{e\in[A,\bar{A}]} \alpha(e) = 0 \qquad \ldots(iii)$$

Since the values of the edges inside A cancel out each other.

Note: If G has nowhere zero k-flow for some k, then G has no bridges.

for example, the graph fig 13.18 is with a nowhere zero 4-flow.

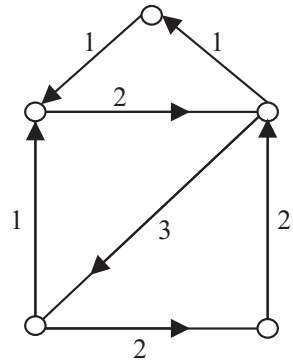


Fig. 13.18

13.12 Tutte's Problem

This problem was conjected by TUTTE(1954). That every bridgeless graph has a nowhere zero 5-flow but does not have any nowhere 4-flows, and so 5 is the best one can think f. Tutte's conjecture resembles the 4-colour theorem, and indeed, the conjective is known to hold for the planar graphs. This proof uses the 4-colour theorem, which will be discussed later on.

Proposition 13.1: *A graph has a 2-flow if and only if all its degrees are even.*

Proof: Since a multigraph admits a k-flow iff it admits a Z_k-flow.

A graph $G = (V, E)$ has 2-flow iff it has a Z_2-flow *i.e.* iff the constant map.

$$\vec{E} \to Z_2 \text{ with value } \overline{1} \text{ satiesfies } (F_2).$$

This is the case iff all degrees are even.

Proposition 13.2: *A cubic graph has a 3-flow iff it is bipertite.*

Proof: Let $G = (V, E)$ be a cubic graph.

We assume that G has a 3-flow, and hence a Z_3-flow f.

We can show that cycle $C = x_0x_1...x_px_o$ in G has been length.

We consider two consicutive edges on C

Say $C_{i-1} := x_{i-1}$.

If f assigned the same value to these edges in the direction of the foreward orientation of C, *i.e.* $f(e_{i-1}, x_{i-1}, x_i) = f(e_i, x_i, x_{i+1})$, than f could not satisfy (F_2) at x_i for any non-zero value of the third edge at x_i.

Therefore, f assign the value $\overline{1}\,\overline{2}$ and to the edges of C alternately, and in particular C has even length.

Conversely, let G is bipartite, with vertex bipartition $\{X, Y\}$. Since G is cubic, the mapping $\vec{E} \to Z_3$ defined by

$f(e, x, y) := \overline{1}$ and $f(e, y, x) := \overline{2}$ for all edges $e = xy$ with $x \in X$ and $y \in Y$ is a Z_3-flow on G.

Hence G has a 3-flow. ■

Proposition 13.3: *For all even $n > 4$, $\phi(K^n) = 3$.*

Proof: $\phi(K^n) \geq 3$ for even n.

By induction process on n, we show that every $G = K^n$ with even $n > 4$ has a 3-flow.

Let $n = 6$, there G is the edge-disjoint union of three graphs G_1, G_2, G_3, with G_1, $G_2 = K^3$ and $G_3 = K_{3,3}$.

It is clear that G_1 and G_2 each have a 2-flow, whicle G_3 has a 3-flow. The union of all these flows is a 3-flow on G.

Now let $n > 6$, and assume the assertion holds for $n - 2$, clearly, G is the edge-disjoint union of a K^{n-2} and a graph $G' = (V', E')$ with $G' = K^{n-2} * K^2$.

The K^{n-2} has a 3-flow by induction.

To find a Z_3-flow on G'.

For every vertex z of the $K^{n-2} \subseteq G'$,

Let f_3 be a Z_3-flow on the triangle $zxyz \subseteq G'$,

where $e = xy$ is the edge of the K^2 in G'.

Let $f: \overline{F}_1 \to Z_3$ be the sum of these flows.

It is clear that, f is nowhere zero, except possible in (e, x, y) and (e, y, x).

If $f(e, x, y) \neq \overline{0}$ then f is the desired Z_3-flow on G'. If $f(e, x, y) = 0$, then $f + f_z$ is a Z_3-flow on G'. ■

Proposition 13.4: *Every 4-edge-connected graphs has a 4-flow.*

Proof: Let G be 4-edge connected graph. G has two edge-disjoint spanning tree T_i, $i = 1, 2$.

For each edge $e \notin T_i$, let $C_{i,e}$ be the unique cycle in T_{i+e} and let $f_{i,e}$ be a $\boldsymbol{Z}_4$-flow of value $\overline{i}$ around $C_{i,e}$ more precisely: a Z_4-circulation on G with values $\overline{i}$ and $-\overline{i}$ on the edges of $C_{i,e}$ and zero otherwise.

Let $$f_i := \sum_{e \in T_1} f_{i,e} \qquad ...(i)$$

Since each $e \notin T_1$ lies only one cycle C_1, e', f_1 takes only the value $\overline{1}$ and $-\overline{1}$ $(= \overline{3})$ outside T_1.

Let $$F := \{e \in E(T_1) \mid f_1(e) = o) \qquad ...(ii)$$

and $$f_2; = \sum_{e \in F} f_{2,e} \qquad ...(iii)$$

As above,

$$f_2(e) + \overline{2} = -\overline{2} \quad \forall\ e \in F \qquad ...(iv)$$

Now, $f := f_1 + f_2$ is the sum of Z_4. Circulations, and hence itself a Z_4 circulation.

Moreover, f is nowhere zero: on edges in F it takes the value $\overline{2}$, on edges of T_1–F it agrees with f_1, and on all edges outside T_1 it takes one of the values $\overline{1}$ or $\overline{3}$.

Hence f is a Z_4-flow on G. ■

SUMMARY

1. A **diagraph** $D = (V, E)$ consists of a finite set of vertices V joined by a set of directed edges E. A directed edge from vertex x to vertex y is denoted by (x, y).

2. The **indegree** of a vertex v, denoted by indeg (v), is the number of edges terminating at v; its **outdegree**, denoted by outdeg (v), is the number of edges leaving v. A vertex with indegree 0 is a **source**; a vertex with outdegree 0 is a **sink.**
3. The **adjacency matrix** of a diagraph is $A = (a_{ij})$, where a_{ij} = number of directed edges from vertex v_i to vertex v_j
4. Let D be a diagraph with vertices $v_1, v_2, ..., v_n$ and e edges. Then

$$\sum_{i=1}^{n} \text{indeg}(v_i) = e = \sum_{i=1}^{n} \text{outdeg}(v_i)$$

5. A loop-free digraph with exactly one edge between any two distinct vertics is a **dominance diagraph** or a **tournament.**
6. A **weighted digraph** has a weight w for every edge.
7. The **weight** of a directed path is the sum of the weights of the edges along the path.
8. The **weighted adjacency matrix** of a digraph (V, E) is $W = (w_{ij})$, where

$$\begin{cases} \infty & \text{if } (v_i, v_j) \notin E \\ \text{weight of edge } (i, j) & \text{otherwise} \end{cases}$$

9. A **shortest path** from u to v in a weighted digraph weighs the least.
10. Dijkstra's algorithm finds a shortest path and its length from the source to any vertex in a weighted digraph.

EXERCISE

1. Show that $\sum_{v \in V} d^-(v) = \hat{I} = \sum_{v \in V} d^+(v)$.
2. Show that is diconnected if and only if D is connected and each block of D is disconnected.
3. Show that G has an orientation D s.t. $|1d^+(v) - d^-(v)| \leq 1$ for all $v \in V$.
4. Show that if D is strict and max $\{\delta^- \delta^+\} \setminus k > 0$, then D contain a directed cycle of length at least $k + 1$.
5. Prove that every tournament has a directed Hamilton path.

6. Prove that $D(G)$ is k-are-connected iff G is k-edge-connected.

7. Find a circular sequence of seven 0'S and seven 1's such that all 4-digit binary numbers except 0000 and 1111 appear as blocks of the sequence.

8. Show, by considering the Petersen graph, that the following statement is false: every graph G has an orientation in which, for every $S \subseteq V$, the cordinalities of $(S, \overline{S})$ and $(\overline{S}, S)$ differ by atmost one.

9. Apply the method of ranking.

(a) the four tournaments shown in Fig. below.

(b) the tournament with adjacency matrix.

	A	B	C	D	E	F	G	H	I	J
A	0	1	1	1	1	1	0	0	1	1
B	0	0	1	0	0	1	0	0	0	0
C	0	0	0	0	0	0	0	0	0	0
D	0	1	1	0	1	1	0	0	1	0
E	0	1	1	0	0	0	0	0	0	0
F	0	0	1	0	1	0	0	0	0	0
G	1	1	1	1	1	1	0	0	1	0
H	1	1	1	1	1	1	1	0	1	1
I	0	1	1	0	1	0	0	0	0	0
J	0	1	1	1	1	1	1	0	1	0

10. (a) Show that Nash-Williams' theorem is equivalent to the following statement: if every bond of G has at least $2k$ edges, then there is an orientation of G in which every bond has at least k arcs in each direction.

(b) Show, by considering the Grotzsch graph, that the following analogue of Nash-Williams' theorem is false: if every cycle of G has at least $2k$ edges, then there is an orientation of G in which every cycle has at least k arcs in each direction.

11. Let $v_1, v_2 ..., v_v$ be the vertices of a digraph D. The adjacency matrix of D is the $v \times v$ matrix $A = [a_{ij}]$ in which a_{ij} is the number of arcs of D with tail v_i and head v_j. Show that the (i, j)th entry of A^k is the number of directed (v_i, v_j)-walks of length k in D.

12. Let $D_1, D_2,...,D_m$ be the components of $\hat{D}$. The condensation $\hat{D}$ of D is a directed graph with m vertices $w_1, w_2,..., w_m$; there is an arc in $\hat{D}$ with tail w_i and head w_j if and only if there is an arc in D with tail in D_i and head in D_j. Show that the condensation $\hat{D}$ of D contains no directed cycle.

13. In a network N, Prove that the value val(f) of a maximum flow equals the capacity a[S] of a minimum cut.

14. State ant prove seymour's 6-flows theorem.

15. Prove that a connected graph G has a flow number $f(G) = 2$ if and only if it is eulerian.

Suggested Readings

1. J.A.Bondy and U.S.R. Murty, *Graph Theory with Applications*, Elsevier, New York, 1976.

2. T.H.Cormen, *Introduction to Algorithms,* McGraw-Hill, New York, 1990.

3. L. Ford and D. Fulkerson, *Flows and Networks*, Prinecton University Press, Princeton, NJ, 1962.

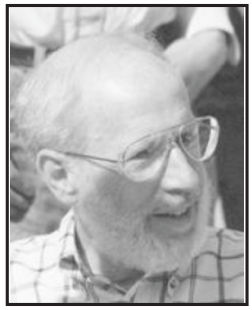
4. R.P. Girmaldi, *Discrete and Combinatorial Mathematics*: *An Applied Introduction*, 4th ed., Addison-Wesley, Reading, MA, 1999, pp. 324-480.

5. E. Horowitz and S.Sahni, *Fundamentals of Data Structures,* Computer Science Press, Potomac, MD, 1976 pp. 301-334.

❑❑❑

14

Matching and Covering



Joseph Bernard Kruskal (1928–)

Joseph Bernard Kruskal (1928-) was born in New York city. He graduated from the university of Chicago in 1948 and received his Ph.D. from Princeton in 1954. He joined as an instructor at Princenton later on he became an assistant professor at the university of Wisconsin and shifted to the university of Michigan in 1958.

He joined Bell Telephone labs in 1959, a position he still holds. He has served as visiting professor at Yale, Columbia and Rutgers.

Kruskal is known for his algorithm in Graph theory which is used to solve so many problems of matching and covering.

14.1 Introduction

If we are given a graph and are asked to find in it as many independent edges as possible. How would we go about this? Will we be able to pair up all its vertices in this way? If not, how can we be sure that this is indeed impossible? Somewhat surprisingly, this basic problem does not only lie at the heart of numerous applications, it also gives rise to some rather interisting graph theory.

A subset M of E is called a matching in G, if its elements are links and no two are adjacent in G; the two ends of an edge is M are called matched under M. A matching M saturates a vertex v, and v is said to be M-saturated, if some edge of M is incident with v; otherwise, v is M-unsaturated. If every vertex of G is M-saturated, the matching M is perfect. M is a maximum matching if G has no matching M' with $|M'|>|M|$;

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_14

It is clear that every perfect matching is maximum. Maximum and perfect matching in graph are indicated in Fig. 14.1. (*a* and *b*).

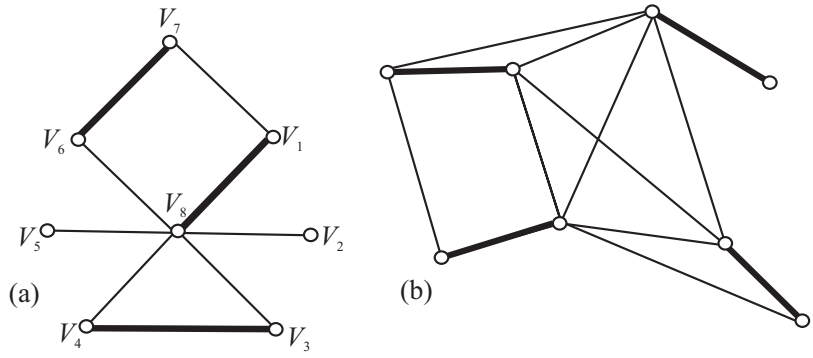


Fig. 14.1 (*a*) A Maximum Matching; (*b*) a Prefect Matching

Let M be a matching in G. An M-alternating path in G is a path whose edges are alternately in $E \setminus M$ and M. In Fig. 14.1(a), the path $v_5\ v_8\ v_1\ v_7\ v_6$ in the graph is an M-alternating path.

An M-augmenting path is an M-alternating path whose origin and terminus are M-unsaturated.

Frankly speaking, a set M of independent edges in a graph $G = (V, E)$ is called a matching. M is a matching of $U \subseteq V$ if every vertex in U is incident with an edge in M. The vertices in U are then called matched (by M); vertices not incident with any edge of M are unmatched.

A k-regular spanning subgraph is called a k-factor. Thus, a subgraph $H \subseteq G$ is a 1-factor of G if and only if $E(H)$ is a matching of V. The problem of how to characterize the graphs that have a 1-factor, *i.e.* a matching of their entire vertex set, will be our main theme in this chapter.

Theorem 14.1: ***(Berge Theorem, 1957)***

A matching M in G is a maximum matching, if G contains no M-augmenting path.

Proof: We consider a matching M in a graph G, and suppose G contains an M-augmenting path $v_0\ v_1\ v_2,..., v_{2m+1}$

Redefining $M' \subseteq E$ by

$M' = (M \setminus \{v_1, v_2, v_3\ v_4,...,v_{2m-1}, v_{2m}\} \cup \{v_0, v_1, v_2\ v_3,..., v_{2m}\ v_{2m-1}\}$

Then M' is a matching in G, and $|M'| = |M| + 1$

Thus M is not a maximum matching.

Conversely, we consider that M is not a maximum matching, and let M' be a maximum matching in G.

Then, $|M'| > |M|$ ...(i)

Set $H = G\,[M \Delta M']$, where $M \Delta M'$ denotes the symmetric difference of M and M'.

Each vertex of H has degree either one or two in H, since it can be incident with atmost one edge of M and one edge of M'. Thus each component of H is either an even cycle with edges alternately in M and M', or else a path with edges alternately in M and M'. from equ.(i), H contains more edges of M' than of M, and therefore some path component P of H must start and end with edges of M'. The origin and terminus of P, being M'-saturated in H, are M-unsaturated in G. Thus we can say that P is an M-augmenting path in G.

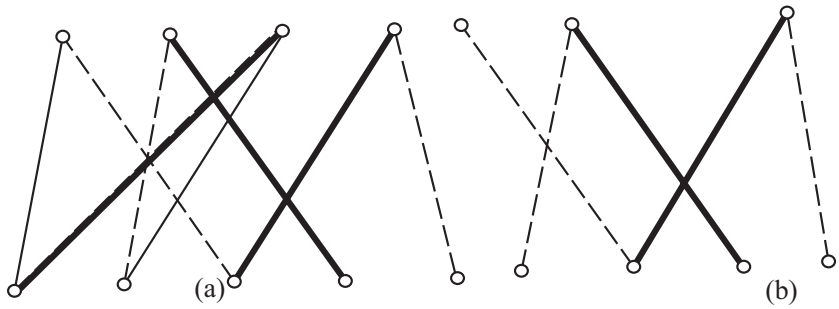


Fig. 14.2: (*a*) G, with M heavy and M´ broken (*b*) G[MΔM´]

14.2 Matching and Covering in Bipertite Graphs

For any set S of vertices in G, we define the neighbour set of S in G to be the set of all vertices adjacent to vertices in S; this set is denoted by $N_G(S)$. Suppose, now, that G is a bipartite graph with bipartition (X, Y). In many applications one wishes to find a matching of G that saturates very vertex in X; an example is the personnel assignment problem. Necessary and sufficient conditions for the existence of such a matching were first given by Hall (1935).

For this whole section, let $G = (V, E)$ be a fixed bipartite graph with bipartition $\{A, B\}$. Vertices denoted as a, a' etc. will be assumed to lie in A, vertices denoted as b etc. will lie in B.

How can we find a matching in G with as many edges as possible?

Let us start by considering an arbitrary matching M in G. A path in G which starts in A at an unmatched vertex and then contains, alternately, edges from $E \setminus M$ and from M, is an alternating path with respect to M. An alternating path P that ends in an unmatched vertex of B is called an augmenting path (Fig. 14.3), because we can use it to turn M into a larger matching: the symmetric difference of M with $E(P)$ is again a matching (consider the edges at a given vertex), and the set of matched vertices is increased by two, the ends of P.

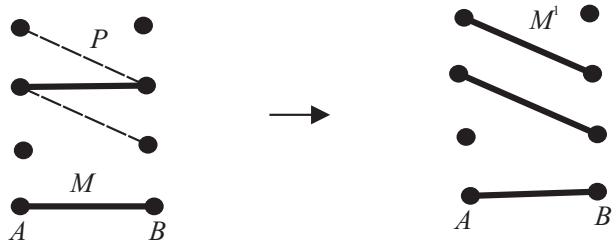


Fig. 14.3: Augmenting the Matching ***M*** by the Alternating Path ***P***

Alternating paths play and important role in the practical search for large matchings. In fact, if we start with any matching and keep applying augmenting paths until no further such improvement is possible, the matching obtained will always be an optimal one, a matching with the largest possible number of edges. The algorithmic problem of finding such matchings thus reduces to that of finding augmenting paths—which is an interesting and accessible algorithmic problem.

Our first theorem characterizes the maximal cardinality of a matching in G by a kind of duality condition. Let us call a set $U \subseteq V$ a cover of E (or a vertex cover of G) if every edge of G is incident with a vertex in U.

Theorem 14.2: ***(König's Theorem 1931)***

The maximum cardinality of matching in G is equal to the minimum cardinality of a vertex cover.

Proof: Let M be a matching in G of maximum cardinality. From every edge in M let us choose one of its ends: its end in B if some alternating path ends in that vertex, and its end in A otherwise. We shall prove that the set U of these $|M|$ vertices covers G; since any vertex cover of G must cover M, there can be none with fewer than $|M|$ vertices, and so the theorem will follow.

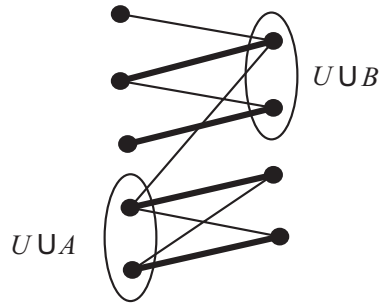


Fig. 14.4: The vertex cover ***U***

Let $ab \in E$ be an edge; we show that either a or b lies in U. If $ab \in M$, this holds by definition of U, so we assume that $ab \notin M$. Since M is a maximal matching, it contains an edge $a'b'$ with $a = a'$ or $b = b'$. In fact, we may assume that $a = a'$; for if a is unmatched (and $b = b'$), then ab is an alternating path, and so the end of $a'b' \in M$ chosen for U was the vertex $b' = b$. Now if $a' = a$ is not in U, then $b' \in U$, and some alternating path P ends in b: either $P' := P'$ (if $b \in P$) or $P' := Pb'a'b$. By the maximality of M, however, P' is not an augmenting path. So b must be matched, and was chosen for U from the edge of M containing it. ■

Theorem 14.3: ***[Hall, 1935] [Marriage Theorem]***

G contains a matching of A if and only if $|N(S) \geq |S|$ for all $S \subseteq A$.

Proof: We give three proofs for the non-trivial implication of this theorem, *i.e.* that the 'marriage condition' implies the existence of a matching of A. The first of these is based on König's theorem; the second is a direct constructive proof by augmenting paths; the third will be an independent proof from first principles.

First proof. If G contains no matching of A, then by Theorem 14.2 it has a cover U consisting of fewer than $|A|$ vertices, say $U = A' \cup B'$ with $A' \subseteq A$ and $B' \subseteq B$. Then

$$|A'| + |B'| = |U| < |A|,$$

and hence

$$|B'| < |A| - |A'| = |A \setminus A'|$$

(Fig. 14.5). By definition of U, however, G has no edges between $A \setminus A'$ and $B \setminus B'$, so

$$|N(A \setminus A')| \leq |B'| < |A \setminus A'|$$

and the marriage condition fails for $S: A \setminus A'$.

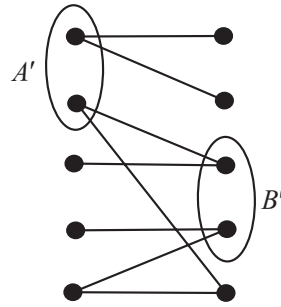


Fig. 14.5: A Cover by Fewer than |***A***| Vertices

Second proof: Consider a matching M of G that leaves a vertex of A unmatched; we shall construct an augmenting path with respect to M. Let $a_0, b_1, a_1, b_2, a_2, \ldots$ be a maximal sequence of distinct vertices $a_i \in A$ and $b_i \in B$ satisfying the following conditions of all $i \geq 1$ (Fig. 14.6):

(i) a_0 is unmatched;

(ii) b_i is adjacent to some vertex $a_{f(|i)} \in \{a_0, \ldots, a_{i-1}\}$;

(iii) $a_i b_i \in M$.

By the marriage condition, our sequence cannot end in vertex of A: that i vertices $a_0, \ldots, a_{i-1}$ together have a least i neighbours in B, so we can always find a new vertex $b_i \neq b_1, \ldots, b_{i-1}$ that satisfies (ii). Let $b_k \in B$ be the last vertex of the sequence. By (i) – (iii),

$$P: = b_k a_{f(k)} b_{f(k)} a_{f2(k)} a_{f3(k)} \ldots a_{fr}(k)$$

with $f^r(k) = 0$

is an alternating path.

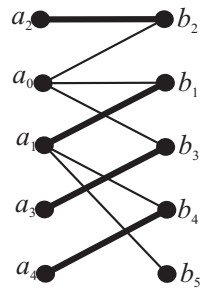


Fig. 14.6: Proving the Marriage Theorem by Alternating Paths

What is it that prevents us from extending our sequence further? If b_k is matched, say to a, we can indeed extend it by setting $a_k := a$, unless $a = a_i$ with $0 < i < k$, in which case (*iii*) would imply $b_k = b_i$ with a contradiction. So b_k is unmatched, and hence P is an augmenting path between a_0 and b_k.

Third proof: We apply induction on $|A|$. For $|A| = 1$ the assertion is true. Now let $|A| \geq 2$, and assume that the marriage condition is sufficient for the existence of a matching of A when $|A|$ is smaller.

If $|N(S)| \geq |S| + 1$ for every non-empty set $S \subset A$, we pick an edge $ab \in G$ and consider the graph $G' := G - \{a, b\}$. Then every non-empty set $S \subseteq A \setminus \{a\}$ satisfies

$$|N_{G'}(S)| \geq |N_G(S) - 1 \geq |S|,$$

so by the induction hypothesis G' contains a matching of $A \setminus \{a\}$. Together with the edge ab, this yields a matching of A in G.

Suppose now that A has a non-empty proper subset A' with $|B'| = |A'|$ for $B' := N(A')$. By the induction hypothesis, $G' := G[A' \cup B']$ contains a matching of A'. But $G - G'$ satisfies the marriage condition too: for any set $S \subseteq A \setminus A'$ with $|N_{G-G'}(S) < |S|$ we would have $|N_G(S \cup A')| < |S \cup A'|$, contrary to our assumption. Again by induction, $G - G'$ contains a matching of $A \setminus A'$. Putting the two matchings together, we obtain a matching of A in G. ■

Corollary 14.1: If $|N(S)| \geq |S| - d$ for every set $S \subseteq A$ and some fixed $d \in N$, then G contains a matching of cardinality $|A| - d$.

Proof: We add d new vertices to B, joining each of them too all the vertices in A. By the marriage theorem the new graph contains a matching of A, and at least $|A| - d$ edges in this matching must be edges of G. ■

Corollary 14.2: If G is k-regular with $k \geq 1$, then G has a 1-factor.

Proof: If G is k-regular, then clearly $|A| = |B|$; it thus suffices to show that G contains a matching of A. Now every set $S \subseteq A$ is joined to $N(S)$ by a total of $k|S|$ edges, and these are among the $k|N(S)|$ edges of G incident with $N(S)$. Therefore $k|S| \leq k|N(S)|$, so G does indeed satisfy the marriage condition. ■

Corollary 14.3: If G is a k-regular bipartite graph with k > 0, then G has a perfect matching.

Proof: Let G be a k-regular bipartite graph with bipartition (X, Y). Since G is k-regular, $k|X| = |E| = k|Y|$ and so, since $k > 0$, $|X| = |Y|$. Now let S be a subset of X and denote by E_1 and E_2 the sets of edges incident with vertices in S and $N(S)$, respectively. By definition of $N(S)$, $E_1 \subseteq E_2$ and therefore

$$k|N(S)| = |E_2| \geq |E_1| = k|S|$$

It follows that $|N(S) \geq |S|$ and hence, by theorem 14.3, that G has a matching M saturating every vertex in X. Since $|X| = |Y|$, M is a perfect matching.

Note: The above theorem is called marriage theorem since it can be more colourfully restated as follows: If every girl in a town knows exactly k boys, and every boy knows exactly k girls, then each girl can marry a boy she knows, and each boy can marry a girl he knows.

14.2.1 Covering

A covering of a graph G is a subset k of V such that every edge of G has at least one and in K. A covering K is a minimum covering if G has no covering K' with $|K'| < |K|$.

If K is a covering of G, and M is a matching of G, then K contains at least one end of each of the edges in M. For any matching M and any covering K, $|M| \leq |K|$.

If M^* is a maximum matching and $\tilde{K}$ is a minimum covering, then

$$|M^*| \leq |\tilde{K}|$$

This result in due to König (1931) and closely related to Hall's Theorem. (see Fig. 14.7).

Lemma 14.1: Let M be a matching and K be covering such that $|M| = |K|$. Then M is a maximum matching and K is a minimum covering.

Proof: If M^* is a maximum matching and $\tilde{K}$ is a minimum covering then,

$$|M| \leq |M^*| \leq |\tilde{K}| \leq |K|$$

since $$|M| = |K|$$

it follows that,

$$|M| = |M^*|$$

and $$|K| = |\tilde{K}|.$$

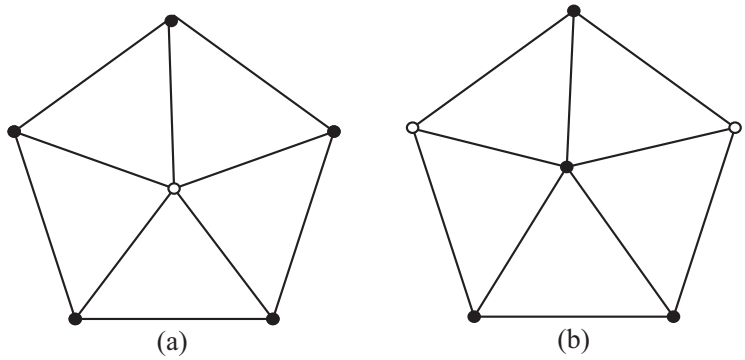


Fig. 14.7: (*a*) A Covering; (*b*) a Minimum Covering

Theorem 14.4: In a bipartite graph, the number of edges in a maximum matching is equal to the number of vertices in a minimum covering.

Proof: Let G be a bipartite graph with bipartition (X, Y), and let M^* be a maximum matching of G. Denote by U the set of M^*-unsaturated vertices in X, and by Z the set of all vertices connected by M^*-alternating paths to vertices of U. Set $S = Z \cap X$ and $T = Z \cap Y$. Then, we have that every vertex in T is M^*-saturated and $N(S) = T$. Define $\tilde{K} = (X \backslash S) \cup T$ (see Fig. 14.8). Every edge of G must have at least one of its ends in $\tilde{K}$. For, otherwise, there would be an edge with one end in S and one end $Y \backslash T$, contradicting $N(S) = T$. Thus $\tilde{K}$ is a covering of G and clearly

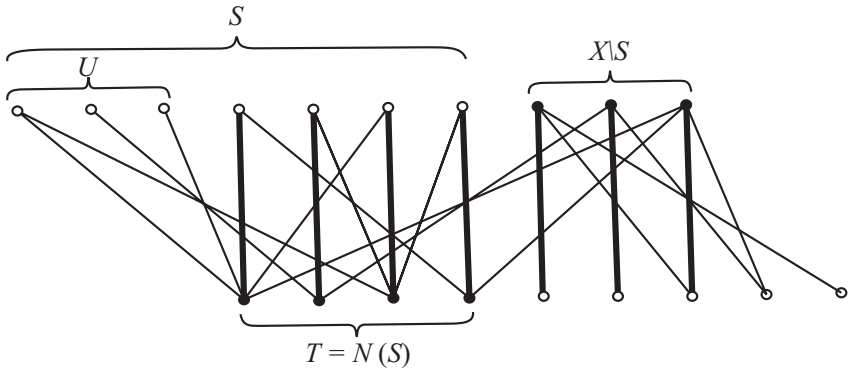


Fig. 14.8

By lemma 14.1, $\tilde{K}$ is a minimum covering, and the theorem follows

14.3 Perfect Matching

A necessary and sufficient condition for a graph to have a perfect matching was obtained by Tutte (1947). A component of a graph is odd or even accordingly as it has an odd or even number of vertices.

Here $0(G)$ is the notation of number of odd components of G.

Theorem 14.5: G has a perfect matching iff $0(G - S) \leq |S|$ for all $S \subseteq V$.

Proof: It is enough to prove this theorem for simple graphs.

Let G has a perfect matching M.

Let S be a proper subset of V.

Let $G_1, G_2, ...G_n$ be the odd components of $G - S$.

Because G_i is odd, some vertex u_i of G_i must be matched under M with a vertex v_i of S. (See Fig. 14.9)

$\therefore$ Since $\{v_1, v_2, ...v_n\} \subseteq S$

$$0(G - S) = n = |\{v_1, v_2, ...v_n\}| \leq |S|$$

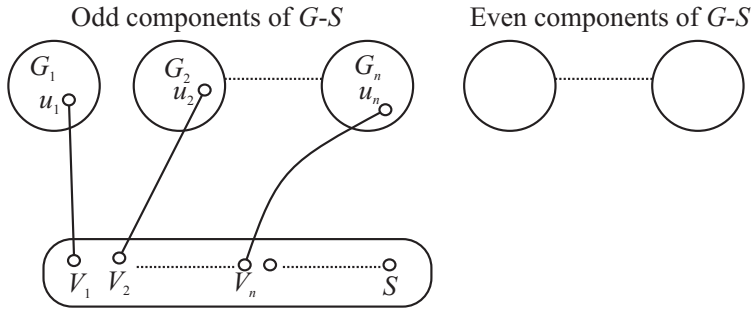


Fig. 14.9 (a)

Conversely,

Suppose that G satisties $0(G - S) \leq |S| \ \forall \ S \subseteq V$ but has no perfect matching.

Let G^* be a maximal graph having no perfect matching.

Then G is a spanning subgraph of G^*.

Since $(G - S)$ is a spanning subgraph of $(G^* - S)$.

We have, $0(G^* - S) \leq 0(G - S)$

By assumption,

$$0(G^* - S) \leq 0(G - S) \leq |S|.$$

for all $S \subseteq V(G^*)$.

In particular, setting $S = \phi$ we have,

$0(G^*) = 0$ and so $V(G^*)$ is even.

Denote by U the set of vertices of degree $(V - 1)$ in G^*

If $U = V$,

then G^* has a perfect matching.

So we assume that $U \neq V$.

Claim: $(G^* - U)$ is a disjoint union of complete graphs.

Suppose that some component of $(G^* - U)$ is not complete.

Then in this component, there are vertices x, y, z ∋

$xy \in E(G^*), yz \in E(G^*)$ and $xz \notin E(G^*)$

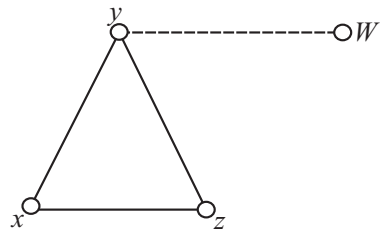


Fig. 14.9 (b)

Moreover, since $y \notin u$, there is a vertex w in $(G^* - U)$ such that $yw \notin E(G^*)$.

Since G* is a maximal graph having no perfect matching,

$(G^* + e)$ has a perfect matching for all $e \notin E(G^*)$.

Let M_1 and M_2 be perfect matchings in $(G^* + xz)$ and $(G^* + yz)$ respectively.

Let H be a subgraph of $G^* U\{xz, yw\}$ induced by $(M_1 \Delta M_2)$.

Since each vertex of H has degree two, H is a disjoint union of cycles.

Also, all of these cycles are even, since edges of M_1 alternate with edges of M_2 around them.

We distinguish two cases.

Case 1: *xz* and *yw* are in different component of *H*.

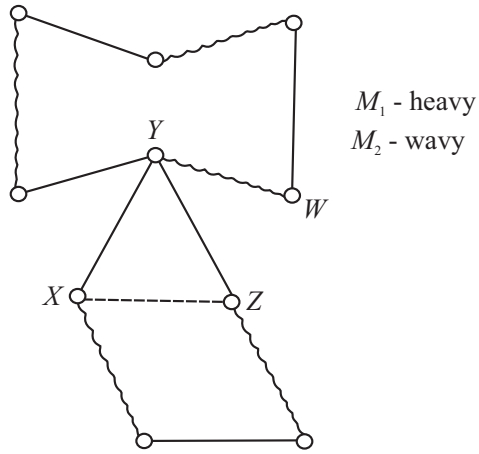


Fig. 14.9 (c)

Then if *yw* is in the cycle *C* of *H*, the edges of M_1 in *C*, together with the edges of M_2 not in *C*, constitute a perfect matching in *G**, contradicting the detinition of *G**.

Case 2: *xz* and *yw* are in the same component of *H*.

By symmetry of *x* and *z*, we may assume that the vertices *x*, *y*, *w* and *z* occur in that order on *C*. [See Fig. 14.9 (*d*)]

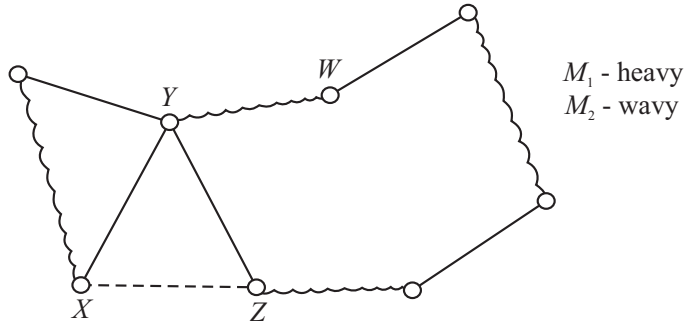


Fig. 14.9 (d)

Then the edges of M_1 in the section *yw ...z* of *c* together with the edge *yz* and the edges of M_2 not in the section *yw ... z* of *c*, constitute a perfect matching in *G**, again contradicting the definition of *G**.

Since both case 1 and case 2 lead to contradictions it follows that (*G** – *U*) is a disjoint union of complete graphs.

By case (1), $0(G^* - U) \leq |U|$

Thus atmost $|U|$ of the components of $(G^* - U)$ are odd.

But then G^* clearly has a perfect matching.

One vertex in each odd component of $(G^* - U)$ is matched with a vertex of U; the remaining vertices in U, and in component of $(G^* - U)$ are then matched as shown in Fig. 14.10.

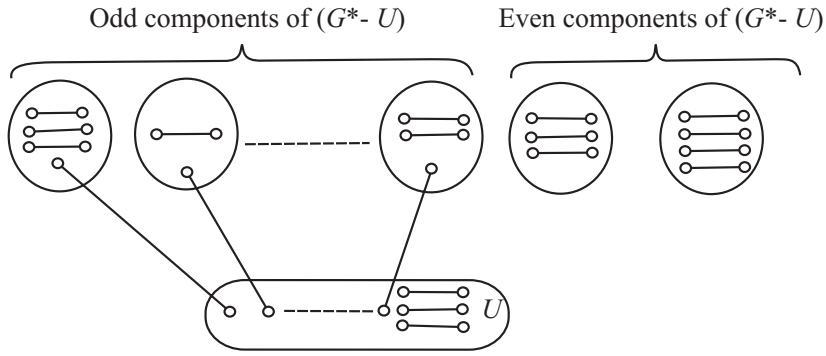


Fig. 14.10

Since G^* was assumed to have no perfect matching. We obtained the desired contradiction.

Thus G have a perfect matching. ■

Corollary 14.1: (*Deduction of Petersen's Theorem from Tutte's Theorem*)

Every 3-regular graph without cut edges has a perfect matching.

Proof: Let G be a 3-regular graph without cut edges and S be a proper subset of V.

We denote by $G_1, G_2, G_3, ..., G_n$ the odd components of $(G - S)$.

Let m_i be the number of edges with one end in G_i and one end is S, $\forall\ 1 \leq i \leq n$.

Since G is 3-regular

$$\sum_{v \in V(G_i)} d(v) = 3v\,(G_i) \ \forall\ 1 \leq i \leq n \qquad ...(i)$$

and

$$\sum_{v \in S} d(v) = 3|S| \qquad ...(ii)$$

from (i)

$$m_i = \sum_{v \in V(G_i)} d(v) - 2 \in (G_i) \text{ is odd}$$

Now $m_i \neq 1$ since G has no cut edge.

Thus, $m_i \geq 3$ for $1 \leq i \leq n$ ...(*iii*)

It follows from (*ii*) and (*iii*) that,

$$0(G - S) = n \leq \frac{1}{3}\sum_{i=1} m_i \leq \frac{1}{3}\sum_{v \in S} d(v) = |S|$$

Hence G has perfect matching.

Note: if G has a 1-factor, then clearly

$0(G - S) \leq |S|$ for all $S \subseteq V|G|$,

since every odd component of $(G - S)$ will send a factor edge to S.

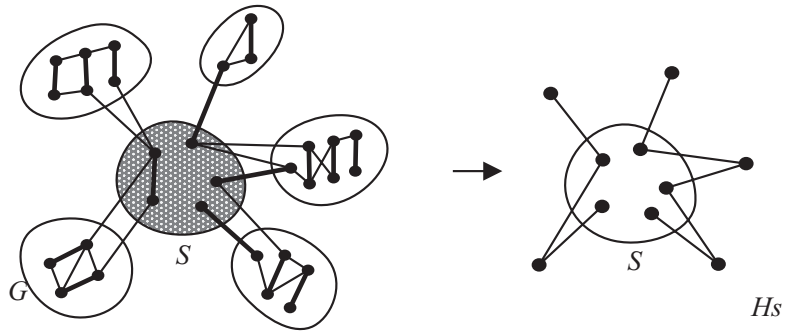


Fig. 14.11: Dednctive Graph

14.4 Factor-critical Graph

A graph $G = (V, E)$ is called factor-critical if $G \neq \phi$ and $(\underline{G} - v)$ has a 1-factor for every vertex $v \in G$. Then G itself has no a-factor because it has odd order.

We call a vertex set $S \subseteq V$ matchable to $(G - S)$ if the graph H_S, which arises from G by contracting the components $C \in C_{G-S}$ to single vertices and deleting all the edges inside S, contains a matching of S. Formally H_S is the graph with vertex set $S \cup C_{G-S}$ and edge set $\{sC \mid \exists\, c \in C : sc \in E\}$ (As shown in Fig. 14.11)

Theorem 14.6: Every graph $G = (V, E)$ contains a vertex set S with the following two properties:

(*i*) S is matchable to $G - S$;

(*ii*) every component of $G - S$ is factor-critical.

Given any such set S, the graph G contains a 1-factor if and only if $|S| = |C_{G-S}|$.

Proof: For any given G, the assertion of Tutte's theorem follows easily from this result. Indeed, by (*i*) and (*ii*) we have $|S| \leq |C_{G-S}| = 0(G - S)$ (since factor-critical graphs have odd order); thus Tutte's condition of $(G - S) \leq |S|$ implies $|S| = |C_{G-S}|$, and the existence of a 1-factor follows from the last statement.

Note first that the last assertion of the theorem follows at once from the assertions (*i*) and (*ii*); if G has a 1-factor, we have $0(G - S) \leq |S|$ and hence $|S| = |C_{G-S}|$ as above; conversely if $|S| = |C_{G-S}|$, then the existence of a 1-factor follows straight from (*i*) and (*ii*).

We now prove the existence of a set S satisfying (*i*) and (*ii*). We apply induction on $|G|$. For $|G| = 0$ we may take $S = \theta$. Now let G be given with $|G| > 0$, and assume the assertion holds for graphs with fewer vertices.

Let d be the least non-negative integer such that

$$0(G - T) \leq |T| + d \text{ for every } T \subseteq V. \qquad ...(i)$$

Then there exists a set T for which equality holds in (1); this follows from the minimality of d if $d > 0$, and from $0(G - \theta) \geq |\theta| + 0$ if $d = 0$.

Let S be such a set T of maximum cardinality, and let $C := C_{G-S}$.

We first show that every component $C \in C$ is odd. If $|C|$ is even, pick a vertex $c \in C$, and let $S' := S \cup \{c\}$ and $C' := C - c$. Then C' has odd order, and thus has at least one odd component. Hence, $0(G - S') \geq 0(G - S) + 1$. Since $T := S$ satisfies (1) with equality, we obtain

$$q(G - S') \geq q(G - S) + 1 = |S| \mid d + 1 = |S'| + d \geq q(G - S') \qquad ...(ii)$$

with equality, which contradicts the maximality of S.

Next we prove the assertion (*ii*), that every $C \in C$ is factor-critical. Suppose there exist $C \in C$ and $c \in C$ such that $C' := C - c$ has no 1-factor. By the induction hypothesis (and the fact that, as shown earlier, for fixed G our theorem implies Tutte's theorem) there exists a set $T' \subseteq V(C')$ with

$$q(C' - T') > |T'|.$$

Since $|C|$ is odd and hence $|C'|$ is even, the numbers $0(C' - T')$ and $[T']$ are either both even or both odd, so they cannot differ by exactly 1. We may therefore sharpen the above inequality to

$$0(C' - T') \geq [T'] + 2.$$

For $T := S \cup \{c\} \cup T'$ we thus obtain

$$\begin{aligned} 0(G - T) &= 0(G - S) - 1 + 0(C' - T') \\ &\geq |S| + d - 1 + |T'| + 2 \\ &= |T| + d \\ &\geq 0(G - T) \end{aligned}$$

with equality, again contradicting the maximality of S.

It remains to show that S is matchable to $G - S$. If $S = \theta$, this is trivial, so we assume that $S \neq \theta$. Since $T := S$ satisfies (1) with equality, this implies that C too is non-empty. $H := H_S$, but 'backwards', *i.e.* with $A := C$. Given $C' \subseteq C$, set $S' := N_H(C') \subseteq S$. Since every $C \in C'$ is an odd component also of $G - S'$, we have

$$|N_H(C')| = |S'| \geq 0(G - S) - d \geq |C'| - d.$$

H contains a matching of cardinality

$$|C| - d = 0(G - S) - d = |S|,$$

which is therefore a matching of S. ■

14.5 Complete Matching

A complete matching of a vertex set V_1 into those in V_2 of the bipertite graph with partition V_1 and V_2 is a matching of the graph in which there is one edge incident with every vertex in V_1. *i.e.* every vertex of V_1 is matched against one vertex is V_2.

The complete matching need not to exist for the entire bipertite graphs. We can observe the fact by considering the interview with large number of suitable candidates for less number of position.

Theorem 14.7: A complete matching of the vertex set V_1 and V_2 in a bipertite graph exists if and only if every subset of r vertices in V_1 is collectively adjacent to r or more vertices in V_2 for all values of r.

Proof: As a homework. (see problems)

Problem 14.1

Find whether a complete matching of V_1 into V_2 exists in the graph (Fig. 14.12)

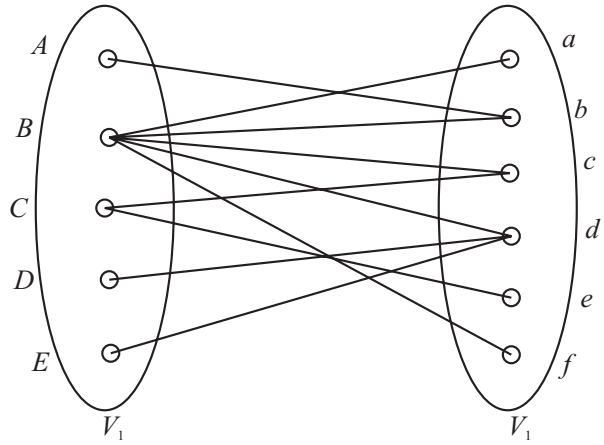


Fig. 14.12

Solution

No, Because we have to take subset $\{D, E\}$ of V_1 having two vertices, then the elements of this set is collectively adjacent to only the subset $\{d\}$ of V_2. The cardinality of $\{d\}$ is one that is less than the cardinality of the set $\{D, E\}$. ■

Problem 14.2

Find whether a complete matching of V_1 into V_2 exist for the following graph? What can you say from V_2 to V_1. (see Fig. 14.13)

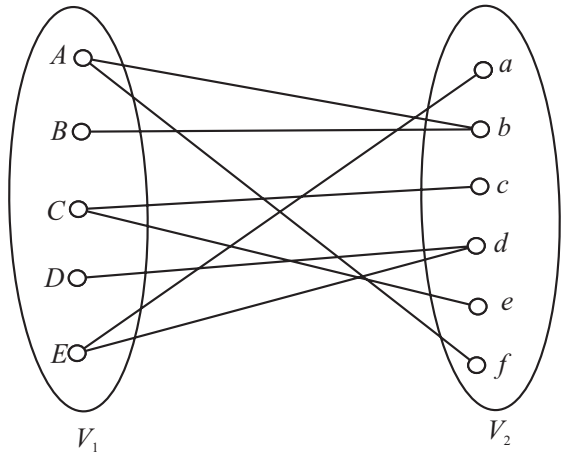


Fig. 14.13

Solution

Yes, we get a complete matching from V_1 into V_2 which is {*Af*, *Bb*, *Cc*, *Dd*, *Ea*}

This matching is not unique, because

{*Af*, *Bb*, *Cc*, *Dd*, *Ea*} is also a complete matching from V_1 into V_2.

Complete matching from V_2 into V_1 does not exist because cardinality of V_2 is more than the cardinality of V_1. ■

Theorem 14.8: For any bipertite graph G with partition V_1 and V_2, if these exists a positive integer M satisfying the condition that $\deg_G(V_1) \geq M \geq \deg_G(V_2)$, for all vertices $v_1 \in V_1$, $v_2 \in V_2$ then a complete matching of V_1 into V_2 exists.

Proof: Let G be a bipertite graph with partition V_1 and V_2.

Let M be a positive integer satisfying the condition that $\deg_G(v_1) \geq M \geq \deg_G(v_2)$, for all vertices $v_1 \in V_1$ & $v_2 \in V_2$ consider a r-element subset S of the set V_1.

Since $\deg(r_1) \geq M$ from each element of S, there are atleast M edges incident to the vertices in V_1. (since no vertex of V_1 is adjacent to the vertex of V_2.

Thus there are $M.r$ edges incident from the set S to the vertices in V_1, but degree of every vertex of V_2 cannot exceed M implies that these $M.r$ edges are incident on atleast

$$\frac{M.r}{M} = r \text{ vertices in } V_2$$

Hence, there exists a complete matching of V_1 into V_2. ■

14.6 Matrix Method to Find Matching of a Bipertite Graph

The adjacency matrix of the bipartite graph G can be written by the rearrangement of rows and columns as

$$X(G) = \begin{bmatrix} 0 & X_{n1\times n2} \\ X^T{}_{n1\times n2} & 0 \end{bmatrix}$$

where $X_{n1 \times n2}$ is (0, 1) whose rows are corresponds the vertices of V_1 and columns corresponds to the vertices of V_2. A matching V_1 into V_2 such that

no row or a column has more than one 1. The matching is complete if the $n_1 \times n_2$ matrix so obtained contains exactly one in each row.

Problem 14.3

Find a complete matching of the graph (Fig. 14.14) by matrix method.

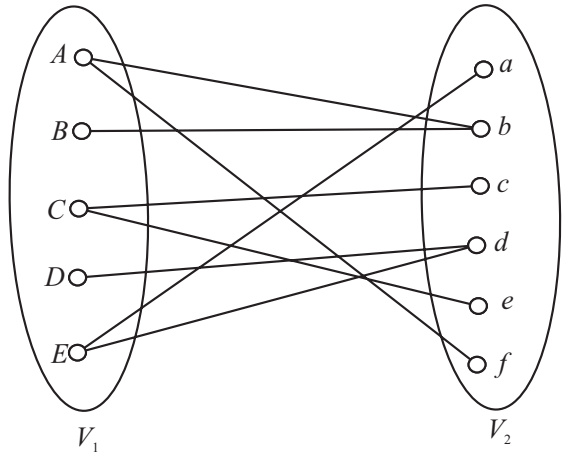


Fig. 14.14

Solution

$$X(G) = \begin{bmatrix} 0 & X_{n1\times n2} \\ X^T{}_{n1\times n2} & 0 \end{bmatrix}$$

Where

$$X_{n1 \times n2} = \begin{array}{c|cccccc} & a & b & c & d & e & f \\ \hline A & 0 & 1 & 0 & 0 & 0 & 1 \\ B & 0 & 1 & 0 & 0 & 0 & 0 \\ C & 0 & 0 & 1 & 0 & 1 & 0 \\ D & 0 & 0 & 0 & 1 & 0 & 0 \\ E & 1 & 0 & 0 & 1 & 0 & 0 \end{array}$$

here $n_1 = 5$, $n_2 = 6$ & $n = n_1 + n_2 = 11$ = total numbers of vertices of G.

***Step 1*:** Choose the row B and the column b.
(since B contains 1 in only one place in the entire now)

***Step 2*:** Discard the column b (since it has already chosen)

Step 3: Choose the row *D* and the column *d* (Since *D* contains 1 in only one place in the entire row)

Step 4: Discard the column *d* (Since it has already chosen)

Step 5: Choose the row *E* and the column a (Since Ea^{th} entry is one and not chosen earlier)

Step 6: Discard the column *a* (chosen in Step 5)

Step 7: Choose the row *A* and column *f* (since the row *A* contains exactly one 1 in the column *f*).

Step 8: Discard the column *f* (chosen in step 7)

Step 9: Choose the row *C* and the column *e* (or *c*) [since *C* is the final (unique) row]

***Step 10*:** Now, no row is left to choose and all the rows are able to be chosen. Hence the matching is complete (if any zero row remains, then the matching is not complete but the process slops)

The resultant matrices after each step and the final matching is given below.

After the steps 1 and 2

	a	b	c	d	e
A	0	0	0	0	1
C	0	1	0	1	0
D	0	0	1	0	0
E	1	0	1	0	0

After the steps 3 and 4

	a	c	e	f
A	0	0	0	1
C	0	1	1	0
E	1	0	0	0

After the steps 5 and 6

	c	e	f
A	0	0	1
C	1	1	0

After the steps 7 and 8

	c	e
C	1	1

Resultant matrix and the corresponding matching are shown in Fig. 14.15

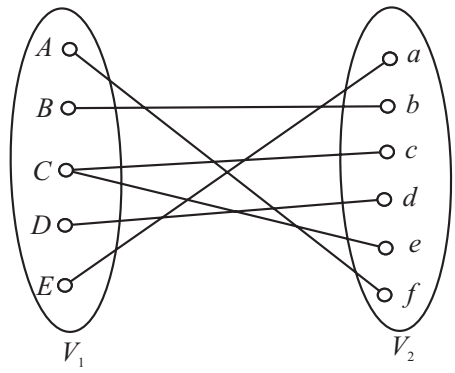


Fig. 14.15

$$\begin{array}{c} \\ A \\ B \\ C \\ D \\ E \end{array}\begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

The complete matching is {Af, Bb, Ce, Dd, Ea}

14.7 Path Covers

In this section we put the above question more generally: how many paths in a given directed graph will suffice to cover its entire vertex set? Of course, this could be asked just as well for undirected graphs. As it turns out, however, the result we shall prove is rather more trivial in the undirected case (exercise), and the directed case will also have an interesting corollary.

A directed path is a directed graph $P \neq \theta$ with distinct vertices $x_0,...,$ x_k and edges $e_0,..,e_{k-1}$ such that e_i is an edge directed from x_i to x_{i+1}, for all $i < k$. We denote the last vertex x_k of P by ter(P). In this section, path will always mean 'directed path'. A path cover of a directed graph G is a set of disjoint paths in G which together contain all the vertices of G. Let us denote the maximum cardinality of an independent set of vertices in G by $\alpha(G)$.

Theorem 14.9: ***(Gallai and Milgram Theorem, 1960)***

Every directed graph G has a path cover by at most $\alpha(G)$ paths

Proof: Given two path covers p_1, p_2 of a graph, we write $p_1 < p_2$ if $\{\text{ter}(P) \mid P \in p_1\} \subseteq \{\text{ter}(P) \mid P \in p_2\}$ and $|p| < |p|_2$. We shall prove the following:

If p is a <-minimal path cover of G, then G contains an independent set $\{v_P \mid P \in p\}$ of vertices with $v_P \in P$ for every $P \in p$.

Clearly, (i) implies the assertion of the theorem.

We prove (i) by induction on $|G|$. Let $p = \{P_1,...,P_m\}$ be given as in (i), and let $v_i := \text{ter}(P_i)$ for every i. If $\{v_i \mid 1 \leq i \leq m\}$ is independent, there is nothing more to show; we may therefore assume that G has an edge from v_2 to v_1. Since $P_2v_2v_1$ is again a path, the minimality of p implies that v_1

is not the only vertex of P_1; let v be the vertex preceding v_1 on P_1. Then $p' := \{P_1v, P_2, ..., P_m\}$ is a path cover of $G' := G - v_1$. We first show that p' is $<$-minimal with this property.

Suppose that $p'' < p'$ is another path cover of G'. If a path $P \in p''$ ends in v, we may replace P in p'' by Pvv_1 to obtain a smaller path cover of G than p, a contradiction to the minimality of p. If a path $P \in p''$ ends in v_2 (but none in v), we replace P in p'' by Pv_2v_1, again contradicting the minimality of p. Hence $\{\text{ter}(P) \mid P \in p''\} \subseteq \{v_3,...,v_m\}$, and in particular $|p''| \leq |p| - 2$. But now p'' and the trivial path $\{v_1\}$ together form a path cover of G that contradicts the minimality of p.

Hence p' is minimal, as claimed. By the induction hypothesis, $\{V(P) \mid P \in p'\}$ has an independent set of representatives. But this is also a set of representatives of p, and (i) is proved.

14.8 Applications

14.8.1 The Personnel Assignment Problem

In a certain company, there are n worker $X_1, X_2, X_3,...,X_n$, available for n jobs $Y_1, Y_2, Y_3,...,Y_n$. Each worker being qualified for one or more of these jobs. The questions arise; can all the men be assigned, one man per job, to jobs for which they are qualified? This problem is called the personnel assignment problem.

We construct a bipartite graph G with bipartition (X, Y), where $X = \{x_1, x_2, ..., x_n\}$, $Y = \{y_1, y_2, ..., y_n\}$ and x_i is joined to y_i if and only if worker X_i is qualified for job Y_j. The problem becomes one of determining whether or not G has a perfect matching. According to Hall's theorem either G has such a matching or there is a subset S of X such that $|N(S) < |S|$. Given any bipartite graph G with bipartition (X, Y), the algorithm either finds a matching of G that saturates every vertex in X or, failing this, finds a subset S of X such that $|N(S)| < |S|$.

We start with an arbitrary matching M. If M saturates every vertex in X, then it is a matching of the required type. If not, we choose an M-unsaturated vertex u in X and systematically search for an M-augmenting path with origin u. Our method of search, to be described in detail below, finds such a path P if one exists; in this case $\hat{M} = M \Delta E(P)$ is a larger matching than M, and hence saturates more vertices in X. We then repeat the procedure with $\hat{M}$ instead of M. If such a path does not exist, the set

Z of all vertices which are connected to u by M-alternating paths is found. Then $S = Z \cap X$ satisfies $|N(S)| < |S|$.

Let M be a matching in G, and let u be an M-unsaturated vertex in X. A tree $H \subseteq G$ is called an M-alternating tree rooted at u if (*i*) $u \in V(H)$, and (*ii*) for every vertex v of H, the unique (u, v)-path in H is an M-alternating path. An M-alternating tree in a graph is shown in Fig.14.17.

The search for an M-augmenting path with origin u involves 'growing' an M-alternating tree H rooted at u. Initially, H consists of just the single vertex u. It is then grown in such a way that, at any stage, either

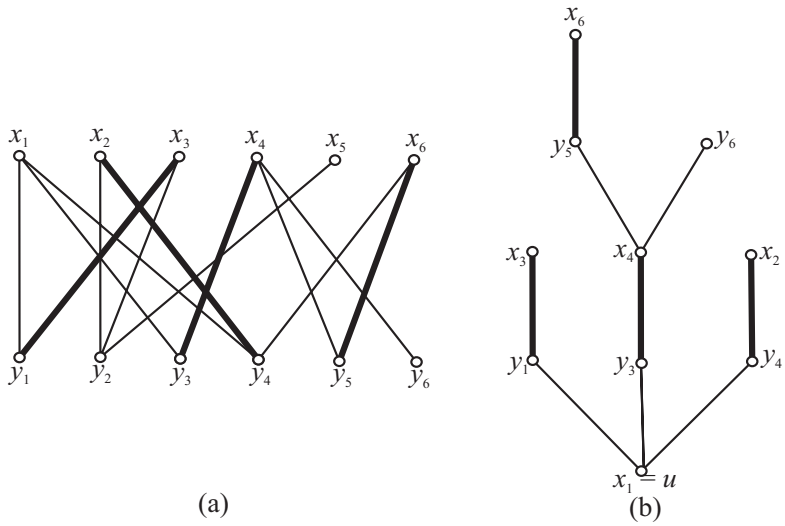


Fig. 14.17: (a) A Matching M in ***G***; (b) an M-alternating Tree in ***G***

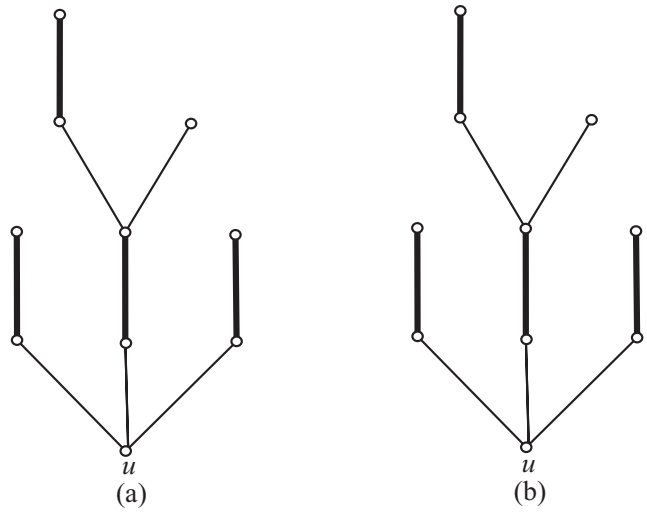


Fig. 14.18: (*a*) Case (***i***) (*b*) Case (***ii***)

(*i*) all vertices of H except u are M-saturated and matched under M [as Fig.14.18 (*a*), or

(*ii*) H contains an M-unsaturated vertex different from u [as in Fig.14.18 (*b*)].

If (*i*) is the case (as it is initially) then, setting $S = V(H) \cap X$ and $T = V(H) \cap Y$, we have $N(S) \supseteq T$; thus either $N(S) = T$ or $N(S) \supset T$.

(*a*) If $N(S) = T$ then, since the vertices in $S \setminus \{u\}$ are matched with the vertices in T, $|N(S) = |S| - 1$, indicating that G has no matching saturating all vertices in X.

(*b*) If $N(S) \supset T$, there is a vertex y in $Y \setminus T$ adjacent to a vertex x in S. Since all vertices of H except u are matched under M, either $x = u$ or else x is matched with a vertex of H. Therefore $xy \notin M$. If y is M-saturated, with $yz \in M$, we grow H by adding the vertices y and z and the edges xy and yz. We are then back in case (*i*). If y is M-unsaturated, we grow H by adding the vertex y and the edge xy, resulting in case (*ii*). The (u, y)-path of H is then an M-augmenting path with origin u, as required.

Figure 14.19 illustrates the above tree-growing procedure.

The algorithm described above is known as the Hungarian method, and can be summarised as follows:

We start with an arbitrary matching M.

1. If M saturates every vertex in X, stop. Otherwise, let u be an M-unsaturated vertex in X. Set $S = \{u\}$ and $T = \theta$.
2. If $N(S) = T$ then $|N(S)| < |S|$, since $|T| = |S| - 1$. Stop, since by Hall's theorem there is no matching that saturates every vertex in X. Otherwise, let $y \in N(S) \setminus T$.
3. If y is M-saturated, let $yz \in M$. Replace S by $S \cup \{z\}$ and T by $T \cup \{y\}$ and go to step 2. (Observe that $|T| = |S| - 1$ is maintained after this replacement.) Otherwise, let P be an M-augmenting (u, y)-path. Replace M by $\hat{M} = M \oplus E(P)$ and go to step 1.

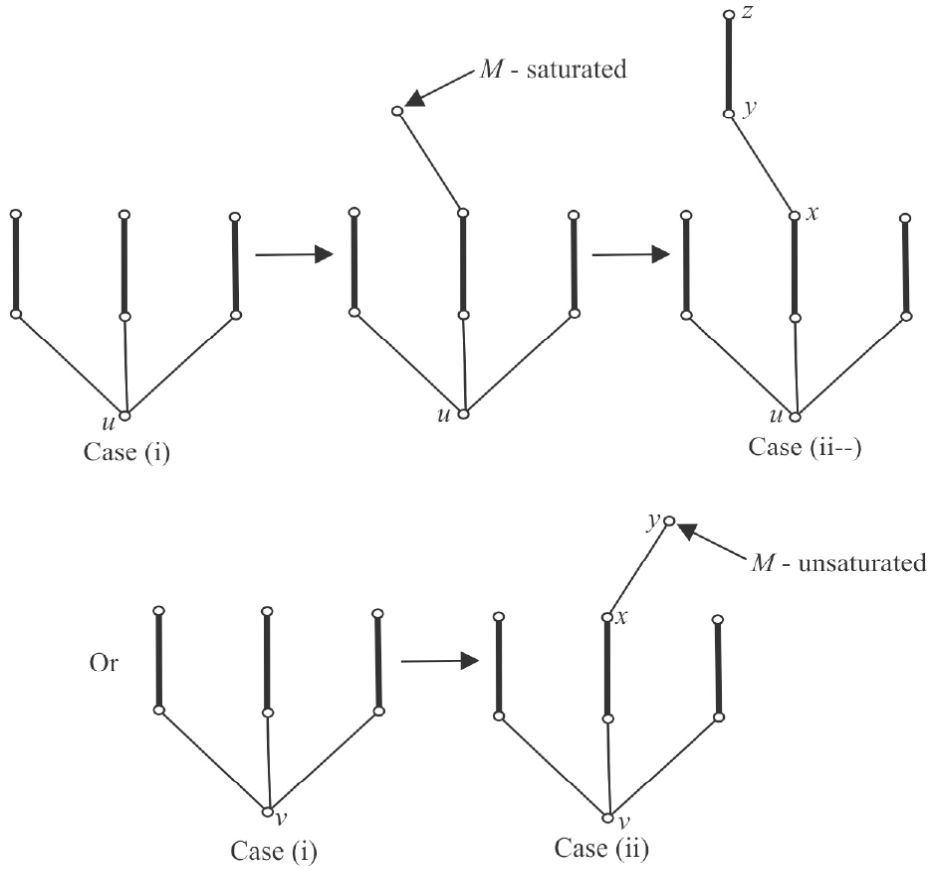
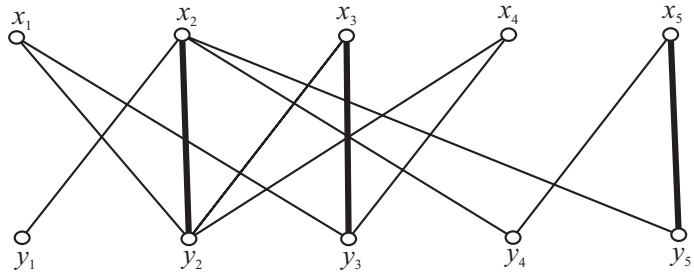


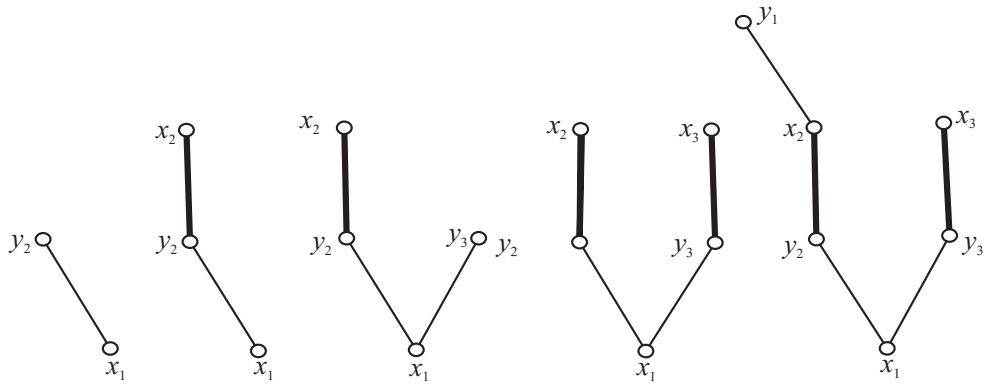
Fig. 14.19: The tree-growing Procedure

Consider, for example, the graph G in Fig. 14.20 (a), with initial matching $M = \{x_2y_2, x_3y_3, x_5y_5\}$. In Fig. 14.20 ($b$) an M-alternating tree is grown, starting with x_1, and the M-augmenting path $x_1y_2x_2y_1$ found. This results in a new matching $\hat{M} = (x_1y_2, x_2y_1, x_3y_3, x_5y_5)$, and an $\hat{M}$ -alternating tree is now grown from x_4 [figures 14.20 (c) and 14.20 (d)] Since there is no $\hat{M}$ -augmenting path with origin x_4, the algorithm terminates. The set $S = \{x_1, x_3, x_4\}$, with neighbour set $N(S) = (y_2, y_3)$, shows that G has no perfect matching.

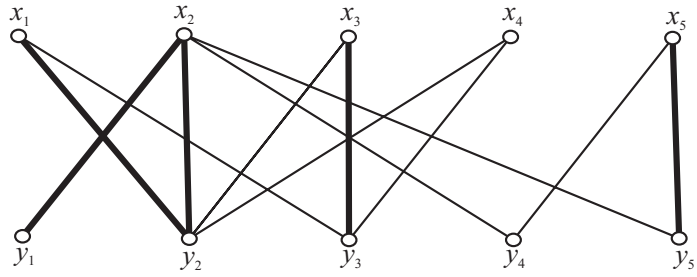
A flow diagram of the Hungarian method is given in Fig.14.21. Since the algorithm can cycle through the tree-growing procedure, I, at most $|X|$ times before finding either an $S \subseteq X$ such that $|N(S)| < |S|$ or an M-augmenting path, and since the initial matching can be augmented at most $|X|$ times before a matching of the required type is found.



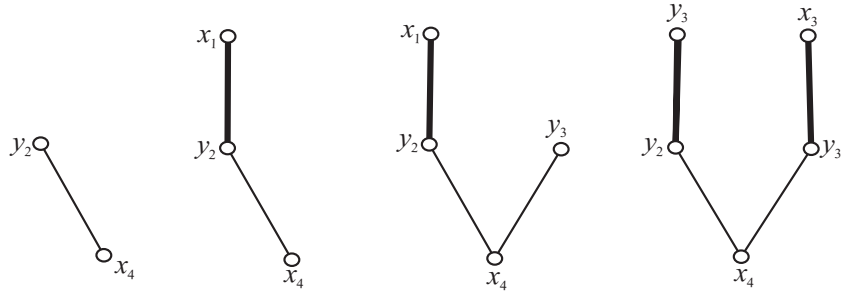
(a)



(b)



(c)



(d)

Fig. 14.20: (*a*) Matching ***M***; (*b*) an ***M***-alternating tree;
(*c*) Matching ***M***; (*d*) an ***M***-alternating tree

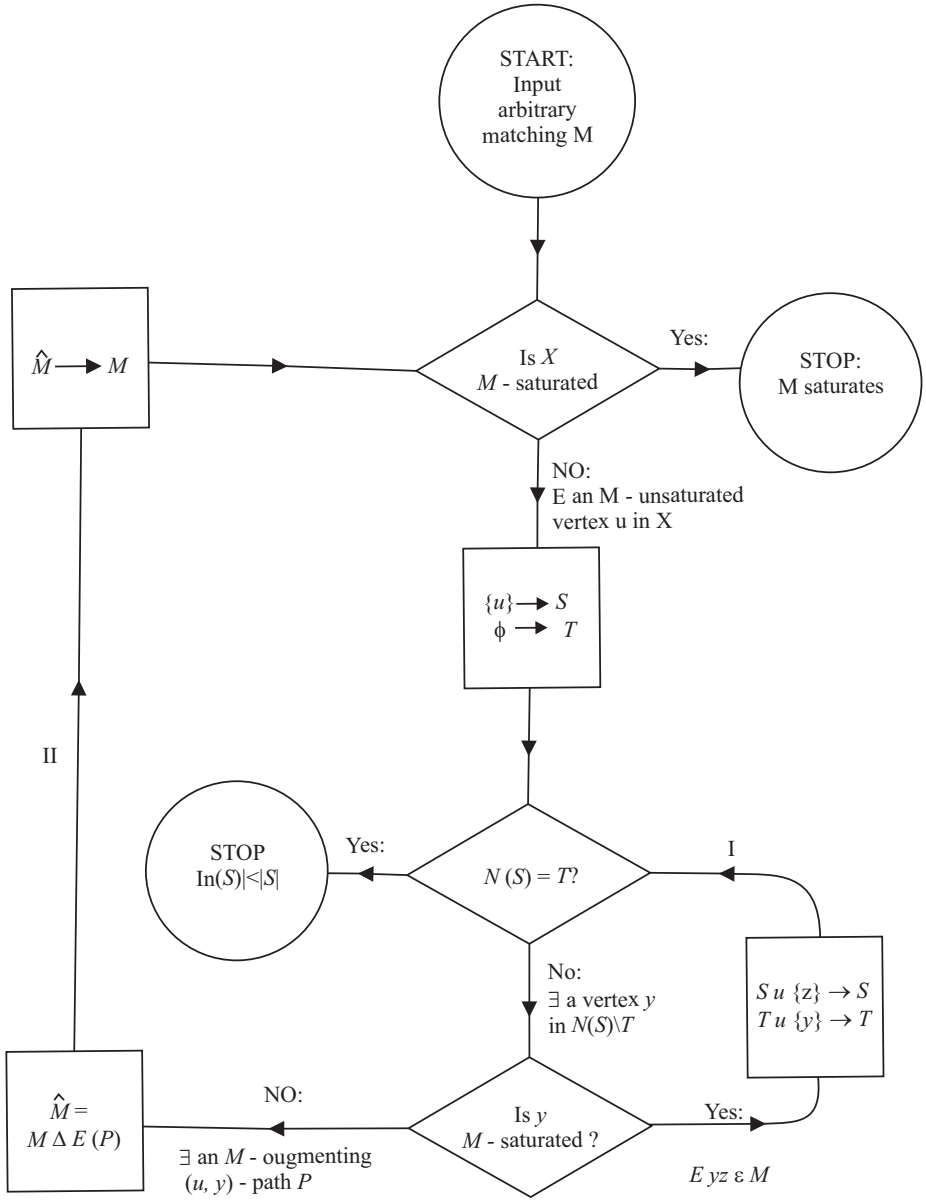


Fig. 14.21

14.8.2 The Optimal Assignment Problem

The Hungariam method is the efficient way to determine a feasible assignment of workers to jobs, if one exists. If we wish to take into account the effectiveness of the workers in their various jobs. On this care we are interested in an assignment that maximises the total effectiveness

of the workers. This problem of finding such an assignment is called the optimal assignment problem.

We consider a weighted complete bipertite graph with bipartition (X, Y), where $X = \{x_1, x_2, x_3..., x_n\}$, $Y = \{y_1, y_2, y_3,..., y_n\}$, and edge $x_i\, y_j$ has weight $w_{ij} = w(x_i.y_j)$, the effectiveness of workers X_i in job Y_i. The optimal assignment problem in clearly equivalent to that of finding a maximum-weight perfect matching in this weighted graph.

We define a feasible vertex labelling as a real-valued function l on the vertex set $X \cup Y$ such that, for all $x \in X$ and $y \in Y$

$$l(x) + l(y) \geq w(xy) \qquad ...(i)$$

(*The real number l(v) is called the lable of the vertex v.*) A feasible vertex labelling is thus a labelling of the vertices such that the sum of the labels of the two ends of an edge is at least as large as the weight of the edge. No matter what the edge weights are, there always exists a feasible vertex labelling; one such is the function l given by

$$\left.\begin{aligned} l(x) &= \max_{y \in y} w(xy) \text{ if } x \in X \\ l(y) &\, 0 \qquad\qquad \text{if } y \in y \end{aligned}\right\} \qquad ...(ii)$$

If l is a feasible vertex labelling, we denote by E_l the set of those edges for which equality holds in (i); that is

$$E_l = \{xy \in E \mid l(x) + l(y) = w(xy)\}$$

The spanning subgraph of G with edge set E_l is referred to as the equality subgraph corresponding to the feasible vertex labelling l, and is denoted by G_i.

14.8.3 Covering to Switching Functions

In logic design of digital machines, we have to minimize the logical functions which are Boolean functions before its implementation, which can be done by using coverings of a graph of Boolean functions. To explain the graph of Boolean function and the procedures of its simplifications, we consider the following Boolean expression.

In Boolean expression, we have P' (or $\overline{P}$) denote NOT P, p.q (or $p \wedge q$) denote p AND q, $p + q$ (or $p \vee q$) denote p OR q.

$$\phi = xyz + x'yz + xy'z + xyz' + x'y'z + xy'z' + x'yz' + x'y'z'$$

Each term correspond to the vertices of a graph of Boolean function and the edges are between those terms that differ by only one variable. The edge between the variables in represented by means of common variables of its end vertices. The graph of *f* is shown in Fig. 14.22.

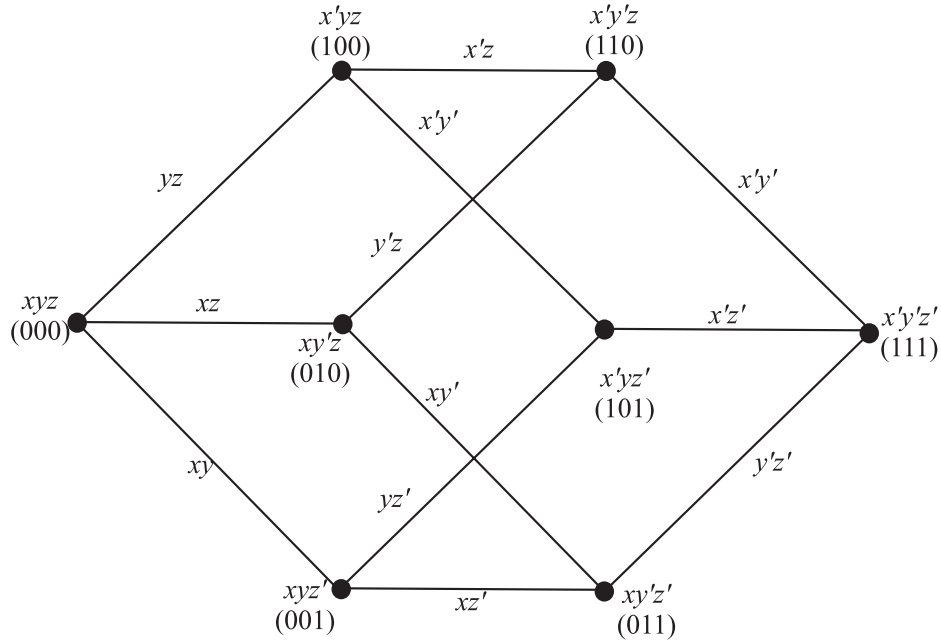


Fig. 14.22 Covering of Switching

One of the covering of the graph is $\{yz, yz'\}$, which is the minimal covering of the graph. Further the graph of expression for $\phi = yz + yz'$ contains only one edge between yz and yz'. This edge is labelled by y and is the minimal covering. Hence the simplified expression for ϕ is $\phi = y$.

Problem 14.4

Minimize the switching function given below using Boolean graph.

$$\phi = w^1x^1y^1z^1 + w^1x^1yz^1 + wx^1y^1z^1 + w^1x^1yz + w^1xyz^1 + wxyz.$$

Solution

The graph of the function f is shown below.

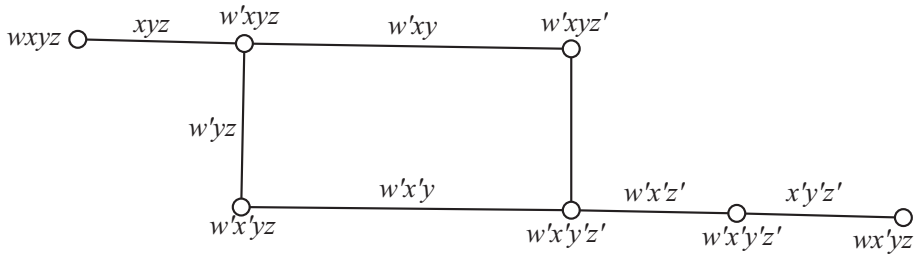


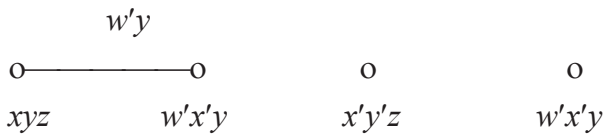
Fig. 14.23

One of the minimal covering of the graph is $\{xyz, x^1y^1z^1, w^1xy, w^1x^1y\}$

$\therefore$ The expression

$\phi = xyz + x^1y^1z^1 + w^1xy + w^1x^1y$ is the simplified expression of ϕ.

To minimize further this expression ϕ, we consider the graph of this function (As above) we get



Now the edge w^1y is the minimal cover of the connected component of this graph. Considering the minimal cover with isolated vertices we get the set $\{w^1y, x^1y^1z^1, w^1x^1y\}$.

Thus the minimized expression for ϕ is

$$\phi = w^1y + x^1y^1z^1 + w^1x^1y.$$

We note that this expression cannot be further simplified as the graph of this function is a totally disconnected graph. ■

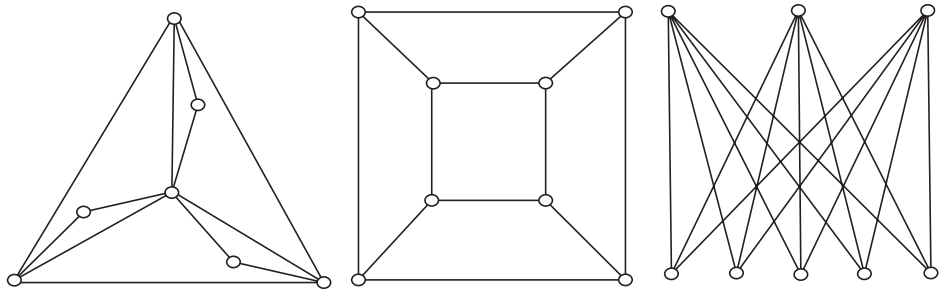
SUMMARY

1. If every vertex in a digraph is reachable from every other vertex, the digraph is **strongly connected.**
2. Let $A = (a_{ij})$ be the adjacency matrix of a digraph D with vertices $v_1, v_2, ..., v_n$. Let $R = A \vee A^{[2]} \vee ... \vee A^{[n-1]} = (r_{ij})$. Then vertix v_j is reachable from vertex v_i if and only if $r_{ij} > 0$, where $i \neq j$. R. is the **reachability matrix** of the digraph.
3. A **de Bruijn sequence** for a binary alphabet is a 2^n-bit sequence that contains every n-bit word as a subsequences in a cyclic fashion.

4. A vertex v_j is **reachable** from a vertex v_i if there is a directed path from v_i to v_j
5. A **dag** is a digraph with no directed cycles.
6. Dags pictorially represent assignment statements and algebraic expressions with repeating subexpressions; they help find the prefix and postfix forms of such expressions.

EXERCISE

1. Show that a tree has at most one perfect matching.
2. Two people play a game on a graph G by alternately selecting distinct vertices v_0, v_1, v_2,..., such that, for $i > 0$, v_i is adjacent to v_{i-1}. The last player able to select a vertex wins. Show that the first player has a winning strategy iff G has no perfect matching.
3. A k-factor of G is a k-regular spanning subgraph of G, and G is k-factorable if there are edge-disjoint k-factors $H_1, H_2,...,H_n$ such that $G = H_1 \dot{\cup} H_2 \dot{\cup} ... \dot{\cup} H_n$.
   (*a*) Show that
      (*i*) $K_{n,n}$ and K_{2n} are 1-factorable;
      (*ii*) the Petersen graph is not 1-factorable.
   (*b*) Which of the following graphs have 2-factors?



4. A non-negative real matrix Q is doubly stochastic if the sum of the entries in each row of Q is 1 and the sum of the entries in each column of Q is 1. A permutation matrix is a (0, 1)-matrix which has exactly one 1 in each row and each column. (Thus every permutation matrix is doubly stochastic.) Show that
   (*a*) every doubly stochastic matrix is necessarily square;

(*b*) every doubly stochastic matrix Q can be expressed as a convex linear combination of permutation matrices, that is

$$Q = c_1P_1 + c_2P_2 + ... + c_kP_k$$

where each P_i is a permutation matrix, each c_i is a non-negative real number, and $\sum_{1}^{k} c_i = 1$.

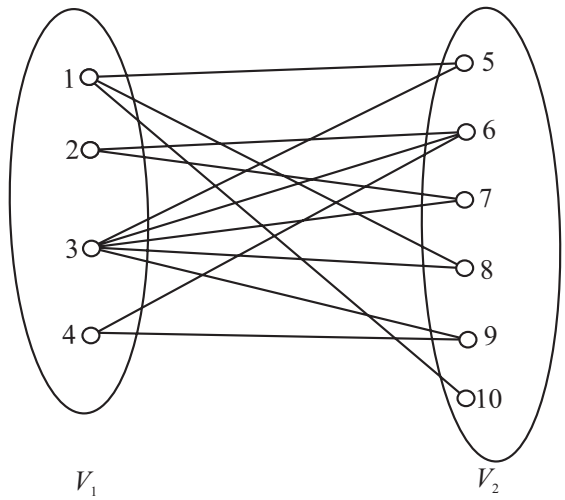
5. Show that a tree G has a perfect matching iff $0(G - v) = 1$ for all $v \in V$.

6. Leg G be simple, with v even and $S < \frac{v}{2}$. Show that if $\in > \left(\frac{\delta}{2}\right) + (v - 2d - 1) + d(v - d)$, then G has a perfect matching.

7. A diagonal of an $n \times n$ matrix is a set of n entries no two of which belong to the same row or the same column. The weight of a diagonal is the sum of the entries in it. Find a minimum-weight diagonal in the following matrix:

$$\begin{bmatrix} 4 & 5 & 8 & 10 & 11 \\ 7 & 6 & 5 & 7 & 4 \\ 8 & 5 & 12 & 9 & 6 \\ 6 & 6 & 13 & 10 & 7 \\ 4 & 5 & 7 & 9 & 8 \end{bmatrix}$$

8. Show that a complete matching of V_1 into V_2 exists in the following graph.



9. Let A be a finite set with subsets $A_1,...,A_n$, and let $d_1,...,d_n \hat{I} N$. Show that there are disjoint subsets $D_k \acute{I} A_k$, with $|D_k| = d_k$ for all k £ n, if and only if-

$$\left| \bigcup_{i \leftarrow I} A_i \right| \geq \sum_{i \in I} d_i$$

for all $I \subseteq \{1,...,n\}$.

10. Find a cubic graph without a 1-factor.

Suggested Readings

1. A.V.Aho *et al, Data Structures and Algorithms*, Addison-Wesley, Reading, MA, 1983 pp. 198-229.

2. R.K.Ahuja *et al.*, *Network Flows,* Prentice-Hall, Englewood Cliffs, NJ, 1993.

3. S.Sahni, *Concepts in Discrete Mathematics*, 2nd ed., Camelot, Fridley, MN, 1985, pp. 379-397.

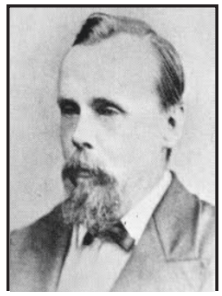
4. R.J.Wilson and J.J. Watkins, *Graphs*: *An Introductory Approach*, Wiley, New York, 1990, pp. 80-111.

5. S.K. Yadav, *Elements of Graph Theory*, Ane Books, New Delhi, 2011.

❑❑❑

15

Colouring of Graphs



Francis Guthrie (1831–1899)

Francis Guthrie (1831–1899) was born in London, England. He graduated from university college London he pursued law and became a barrister. In 1850s, after colouring the countries of England on a map with four colours. He conjectured the four colour problem to his younger brother Frederick. At that time Frederick Guthrie (1833-1866) was a student of De Morgan, had showed the problem to him, who in turn communicated to William Hamilton in 1852. Hamilton did not take interest in this problem. Later on in 1878, the problem was popularized by Arthur Cayley with an announcement in a meeting of the London Mathematical society.

In 1861, Guthrie joined as a professor of mathematics at Graft-Reinet College, South Africa and later on he moved to South African University, Cape Town in 1876 where he served until death.

15.1 Introduction

In graph theory, Graph colouring is a special case of graph labeling. It is an assignment of labels traditionally called "Colour's" to elements of graph subject to certain constraint in. Its simplest form, it is a way of colouring the vertices of a graph such that no two adjacent vertices share the same colour, this is called vertex colouring. Similarly, an edge colouring assigns a colour to each edge so that no two adjacent edges share the same colour, and a face colouring of a planer graph assigns a colour to each face or region so that no two faces that share a boundary have the same colour.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2_15

Vertex colouring is the starting point of the subject and the other colouring problems can be transformed into a vertex version. For example, an edge colouring of a graph is just a vertex colouring of its line graph, and a face colouring of a planar graph is just a vertex colouring of its planar dual. However, non-vertex colouring problems are often stand and studied as it. That is partly for perspective, and partly became some problems are best studied in non-vertex form, as for instance is edge colouring. The convention of using colours originates from colouring the countries or states of a map, where each face is titerally coloured. This was generalized to colouring the faces of a graph embedded in the plane. By planar duality it became colouring the vertices, and in this form it generalizes to all graphs. In mathematical and computer representation it is typical to use the first few positive or non-negative integers as the colours.

Generally we can use any finite set as the colour set. The nature of the colouring problem depend on the number of colours but not on what they are.

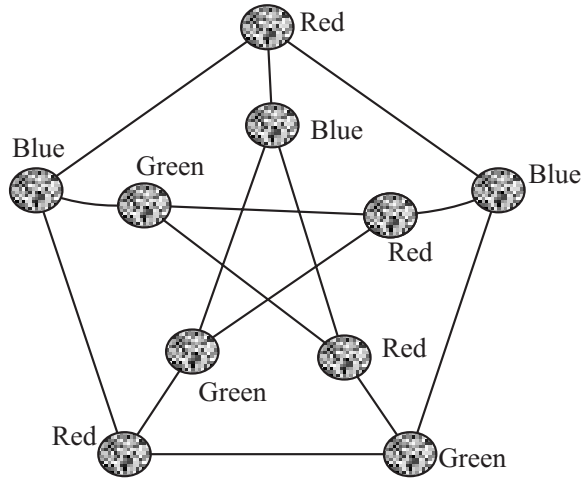


Fig. 15.1: A Proper Vertex Colouring of the Petersen Graph with 3 Colors, the Minimum Number Possible

Colouring of graph enjoys many practical applications as well as theoretical challenges. Beside the classical types of problems, different limitations can be set on the graph, or on the way a colour is assigned, or even on the colour itself. It has reached popularity with the general public in the form of popular number puzzle Sudoku. Graph colouring is very active in the field of research.

The first results about graph colouring deal almost exclusively with planar graphs in the form of the coloring of maps, While trying to color a map of the counties of England. **Francis Guthrie** postulated the four color conjecture, noting that four colors were sufficient to color the map so that no regions sharing a common border received the same color. Guthrie's brother passed on the question to his mathematics teacher **Augustus de Morgan** at University College, who mentioned it in a letter to William Hamilton in 1852. **Arthur Cayley** raised the problem at a meeting of the London Mathematical Society in 1879. The same year, **Alfred Kempe** published a paper that claimed to establish the result, and for a decade the four color problem was considered solved. For his accomplishment Kempe was elected a Fellow of the Royal Society and later President of the London Mathematical Society.

In 1890, Heawood pointed out that Kempe's argument was wrong. However, in that paper he proved the five color theorem, saying that every planar map can be colored with no more than, five colors, using ideas of Kempe. In the following century, a vast amount of work and theories were developed to reduce the number of colors to four, until the four color theorem was finally proved in 1976 by **Kenneth** Appel and **Wolfgang Haken**. Perhaps surprisingly, the proof went back to the ideas of Heawood and Kempe and largely disregarded the intervening developments. The proof of the four color theorem is also noteworthy for being the first major computer-aided proof.

In 1912, **George David** Birkhoff introduced the chromatic polynomial to study the coloring problems, which was generalised to the Tutte polynomial by Tutte, important structures in algebraic graph theory. Kempe had already drawn attention to the general, non-planar case in 1879, and many results on generalisations of planar graph coloring to surfaces of higher order followed in the early 20th century.

In 1960, **Claude Berge** formulated another conjecture about graph coloring, the strong perfect graph conjecture, originally motivated by an information-theoretic concept called the zero-error capacity of a graph introduced by Shannon. The conjecture remained unresolved for 40 years, until it was established as the celebrated strong perfect graph theorem in 2002 by **Chudnovsky**, **Robertson**, **Seymour**. **Thomas**.

Graph coloring has been studied as an algorithmic problem since the early 1970s: the chromatic number problem is one of Karp's 21 NP-complete problems from 1972, and at approximately the same time various

exponential-time algorithms were developed based on backtracking and on the deletioncontraction recurrence of **Zykov** (1949). One of the major applications of graph coloring, register allocation in compilers, was introduced in 1981.

Painting all the vertices of a graph with colours such that no two adjacent vertices have the same colour is called the proper colouring of a graph. A graph in which every vertex has been assigned a colour according to a proper colouring is called a properly coloured graph. (see Fig. 15.1).

15.2 Vertex Colouring

When used without any particular qualification, a colouring of graph is almost always a proper vertex colouring namely a labelling of the graph's vertices with colours such that no two vertices sharing the same edge have the same colour. Since a vertex with a loop could never be properly coloured, it is understood that graphs in this context are loopless. The terminology of using colours for vertex lebels goes back to map colouring. Lebels like red and blue are only used when the number of colours is small, and normally it is understood that the labels are drawn from the integers {1, 2, 3...}.

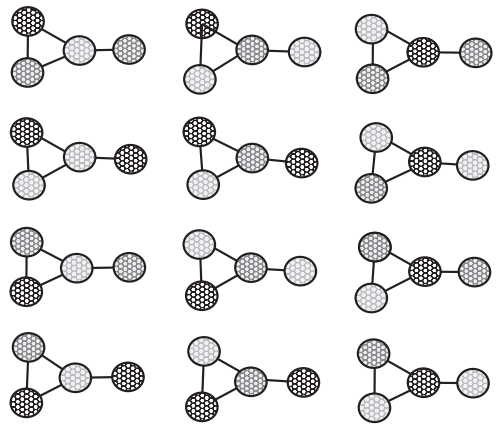


Fig. 15.2: This Graph can be 3-coloured in 12 Different Ways

A colouring using at most k colours is called a proper k-colouring. The smallest number of colours needed to colour a graph G is called needed to colour a graph G is called its chromatic number $X(G)$. A graph that can be assigned a proper k-colouring is k-colourable, and it is k-chromatic of its

chromatic number is exactly k. A subset of vertices assigned to the same colour is called a colour class, every such class forms an independent set. Thus, a k-colouring is the same as a partition of the vertex set into k independent sets, and the terms k-partite and k-colourable have the same meaning.

15.3 Chromatic Polynomial

The chromatic polynomial counts the number of ways a graph can be coloured using no more than a given number of colours. For example, using three colours, the graph in the image to the right can be coloured into 12 ways. With only two colours, it can not be coloured at all. With four colours, it can be coloured in 24 + 4*12 = 72 ways. There are 4! = 24 valid colourings (every assignment of four colours to any 4-vertex graph is a proper colouring) and for every choice of three of the four colours, there are 12 valid 3-colourings. So, for the graph in the example, a table of number of valid colouring would start like this:

Available Colour	1	2	3	4	…
No of Colouring	0	0	12	72	…

The chromatic Polynomial is a function $P(G, t)$ that counts the number of t-colourings of G. As the name indicates, for a given G the function is a polynomial in it. For example the graph, $P(G, t) = t(t-1)^2(t-2)$ and indeed $P(G, 4) = 72$.

The Chromatic Polynomial includes at least as much information about the colourability of G as does the chromatic number. Indeed, X is the smallest positive integer that is not a root of chromatic polynomial.

$$X(G) = \min\{k : P(G, k) > 0\}.$$

Chromatic polynomials for certain graphs are:

Triangle	$K_3\ t(t-1)(t-2)$
Complete graph	$K_n\ t(t-1)(t-2)\ \ldots\ (t-(n-1))$
Tree with n vertices	$t(t-1)^{n-1}$
Cycle	$C_n\ (t-1)^n + (-1)^n(t-1)$
Petersen graph	$t\,(t-1)(t-2)(t^7 - 12t^6 + 67t^5 - 230t^4 + 529t^3 - 814t^2 + 775t - 352)$

15.3.1 Bounds of the Chromatic Number

1. Assigning distinct colours to distinct vertices always yields a proper colouring. So,
$$1 \leq \chi(G) \leq n.$$
2. The only graph that can be 1-coloured are edgeless graphs, and the complete graph K_n of n vertices requires $\chi(K_n) = n$ colours. In the optimal colouring there must be atleast one of the graph's m edges between every pair of colour classes. So,
$$\chi(G)\,[\chi(G) - 1] \leq 2m$$
3. If G contains a clique of size k, then atleast k colours are needed to colour that clique; in other words, the chromatic number is at least the clique number:
$$\chi(G) \geq w(G).$$
for interval of graphs this bound is tight.
4. The two-colourable graphs are exactly the bipertite graphs, including trees and forests. By the four colour theorem, every planar graph can be 4-colurable.
5. A greedy colouring shows that every graph can be coloured with one more colour than the maximum vertex degree, $X(G) \leq \Delta(G) + 1$.
6. Complete graphs have $\chi(G) = n$ and $\Delta(G) = n - 1$, and odd cycles have $\chi(G) = 3$ and $\Delta(G) = 2$ so, for these graphs this bound is best possible. In all other cases, the bound can be slightly improved.

Example 15.1

Find the chromatic polynomial of the graphs given in Fig. 15.3.

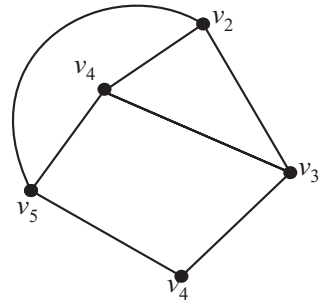


Fig. 15.3: A 3-Cromatic Graph

Solution

We have

$$P_5(\lambda) = C_1\lambda + C_2\frac{\lambda(\lambda-1)}{2!} + C_3\frac{\lambda(\lambda-1)(\lambda-2)}{3!}$$

$$+C_4\frac{\lambda(\lambda-1)(\lambda-2)(\lambda-3)}{4!} + C_5\frac{\lambda(\lambda-1)(\lambda-2)(\lambda-3)(\lambda-4)}{5!}$$

The graph has a triangle, so, it will require at least three different colours for proper colouring.

Therefore, $C_1 = C_2 = 0$ and $C_5 = 5!$ to evaluate C_3, we consider three colours x, y and z These three colours can be assigned properly to vertices v_1, v_2 and v_3 in 3!(= 6) different ways.

After doing that, no more choice is left because v_5 must have the same colour as v_3 and v_4 must have as v_2.

Therefore $C_3 = 6$

Similarly, with four colours v_1, v_2 and v_3 can be properly coloured in 4*6 (= 24) different ways.

The fourth colour can be assigned to v_4 and v_5 with two choices. The fifth vertex provides us additional choice.

Therefore, $C_4 = 24*2 = 48$

making substitution in $P_5(\lambda)$, we get

$$P_5(\lambda) = \lambda(\lambda - 1)(\lambda - 2) + 2\lambda(\lambda - 1)(\lambda - 2)(\lambda - 3) + \lambda(\lambda - 1)(\lambda - 2)(\lambda - 3)(\lambda - 4)$$

$$= \lambda(\lambda - 1)(\lambda - 2)(\lambda^2 - 5\lambda + 7)$$

The presences of factors $\lambda - 1$ and $\lambda - 2$ indicates that G has at least 3-chromatic. ■

Example 15.2

Find the chromatic number for the graph $K_{3,3}$.

Solution

The chromatic polynomial for $K_{3,3}$ is given by $\lambda(\lambda - 1)^5$

Thus the chromatic number of this graph is 2. (see Fig. 15.4)

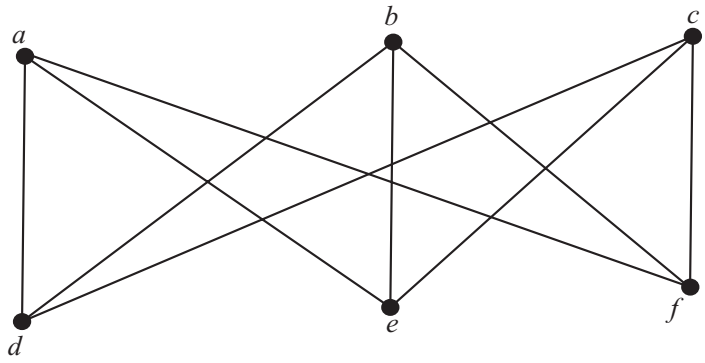


Fig. 15.4

The vertices a, b and c may have the same colour, as they are not adjacent.

Similarly, vertices d, e and f can be coloured in proper way using one colour. ■

But a vertex from $\{a, b, c\}$ and a vertex from $\{d, e, f\}$ both can not have the same colour.

i.e. every chromatic number of any bipertite graph is always 2.

Theorem 15.1: Let $\Delta(G)$ be the maximum of degrees of the vertices of a graph G. Then $\chi(G) \leq 1 + \Delta(G)$.

Proof: The theorem can be proved by induction on V *i.e.* the vertices of the graph.

When $V = 1$, $\Delta(G) = 0$ and $\chi(G) = 1$, (holds)

Let K be an $\forall$ *integer* $K \geq 1$, and assume that the result holds for all graphs with $V = K$ vertices. Suppose G is a graph with $(K + 1)$ vertices

Let v be any vertex of G and let $G_0 = \frac{G}{\{v\}}$ be the subgraph with v deleted.

We note that $\Delta(G_0) \leq \Delta(G)$

Then G_0 has K vertices. We can use the induction hypothesis to conclude that $X(G_0) \leq 1 + \Delta(G_0)$

Then, $\chi(G_0) \leq 1 + \Delta(G_0)$, so G_0 can be coloured with atmost $[1 + \Delta(G)]$ colours.

Since there are atmost $\Delta(G)$ vertices adjacent to v, one of the variable $[1 + \Delta(G)]$ colours remains for v. Thus, G can be coloured with atmost $[1 + \Delta(G)]$ colours. ■

Theorem 15.2: The minimum number of hours for a schedule of committee meetings in one scheduling problem is $\chi(G_0)$.

Proof: We consider $\chi(G_0) = K$.

and the colours used in colouring G_0 are 1, 2, 3, ..., K. firstly we assest that all committees can be scheduled in K one-hour time periods.

We also consider all these vertices coloured 1(say) and the committees corresponding to these vertices. Since no two vertices coloured 1 are adjacent, no two such committees contain the same member.

Hence, all these committees can be scheduled to meet at the same time.

Thus, all committees corresponding to same colour vertices can meet at the same time.

Therefore, all committees can be scheduled to meet during K time periods.

Further, we show that all committees cannot be scheduled in less than K hours. We can prove this by contradiction. Suppose that we can schedule the committees in m one hour time periods, where $K > M$

So, we can give G_0 as m-colouring by colouring with the same colour all vertices which correspond to committees meeting at the same time.

To see that this is a legitimate m-colouring of G_0.

We consider two adjacent vertices.

These vertices correspond to two committees containing one or more common members.

Hence, these committees meet at different times, and thus the vertices are coloured differently.

However, an m-colouring of G_0 gives a contradiction since, we have $\chi(G_0) = K$. ■

15.4 Exams Scheduling Problem

This problem aims "How can the final exams at a university be scheduled so that no student has two exams at the same time?"

This scheduling problem can be solved by using a graph model, with vertex colouring. In this problem vertices represent the course and with an edge between two vertices of there is a common student in the courses they represent. Each time slot for a final exams represented by

different colour. A scheduling of exams corresponds to a colouring of the associated graph.

Suppose these are seven finals to be scheduled

Suppose the colours are numbered 1 through 7 and the following pairs of colours have common students:

1 and 2, 1 and 3, 1 and 4, 1 and 7, 2 and 3, 2 and 4, 2 and 5, 2 and 7, 3 and 4, 3 and 6, 3 and 7, 4 and 5, 4 and 6, 5 and 6, 5 and 7, and 6 and 7. (see Fig. 15.5)

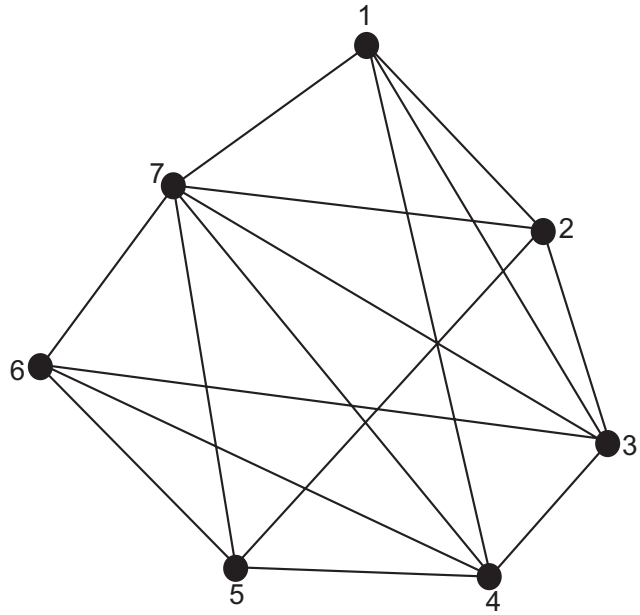


Fig. 15.5: Scheduling Problem

A scheduling consists of a colouring of this graph. Since the chromatic number of this graph is 4 so, four times slots are needed.

A colouring of graph using four colours and the associated schedule are shown in Fig. 15.6.

Time	**Courses**
(I)	1, 6
(II)	2
(III)	3, 5
(IV)	4, 7

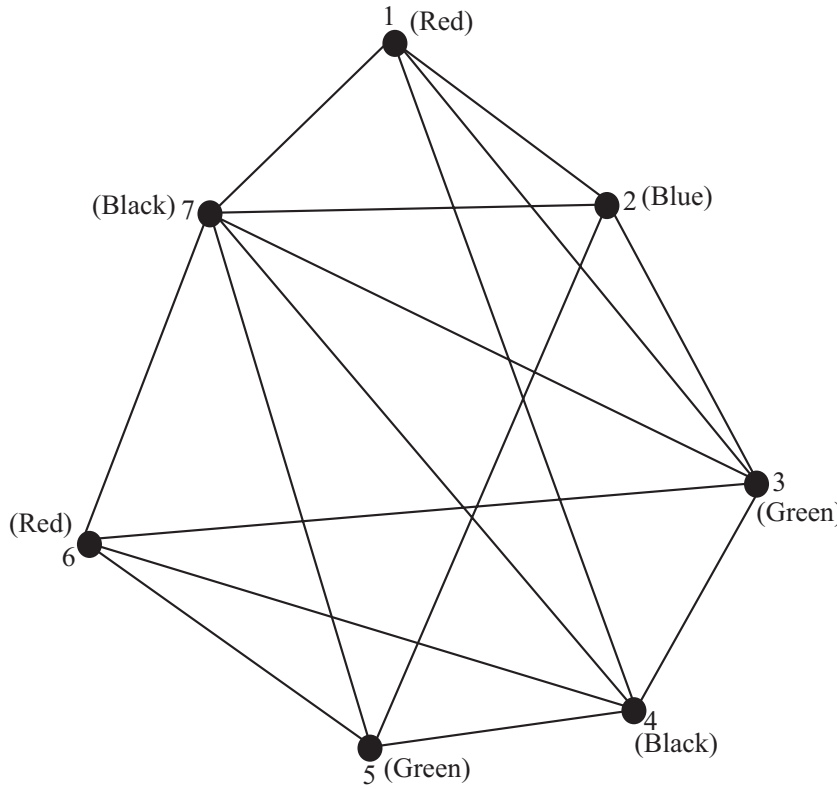


Fig. 15.6: Colouring Schedule

Using colouring to schedule final exams.

Example 15.3

What is the chromatic number of the complete bipertite graph $K_{m, n}$ where m and n are positive integers?

Solution

In this problem, the number of colours needed may seem to depend upon *m* and *n*.

However, only two colours are needed. Colour the set of *m*-vertices with one colour and the set of *n*-vertices with a second colour.

Since edge can not only a vertex from the set of *m*-vertices and a vertex from the set of *n*-vertices, no two adjacent vertices have the same colour.

We can take a colouring of $K_{3, 4}$ with two colours as shown in Fig. 15.7.

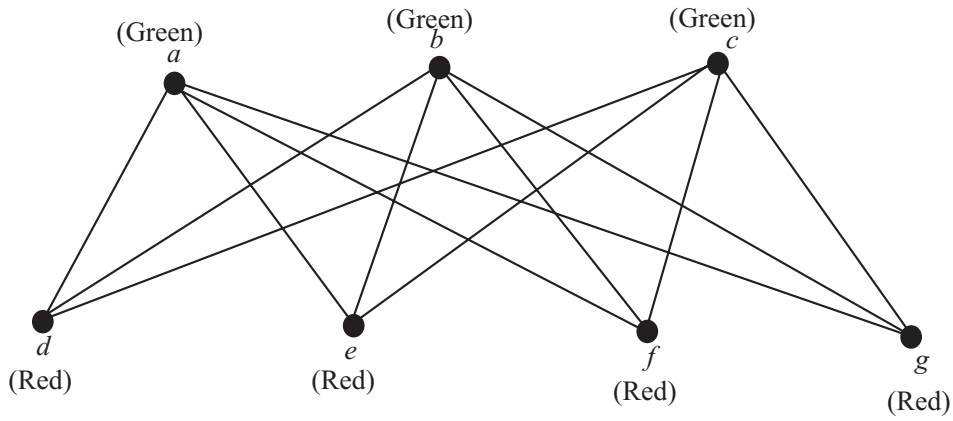


Fig. 15.7: A colouring of $K_{3,4}$.

A colouring of K_n can be constructed using is colours by assigning a different colour to each vertex. There is no colouring of fewer colours. No two vertices can be assigned the same colour, since every two vertices of this graph are adjacent.

Hence, the chromatic number of $K_n = n$.

we can see the colouring of K_5 in Fig. 15.8

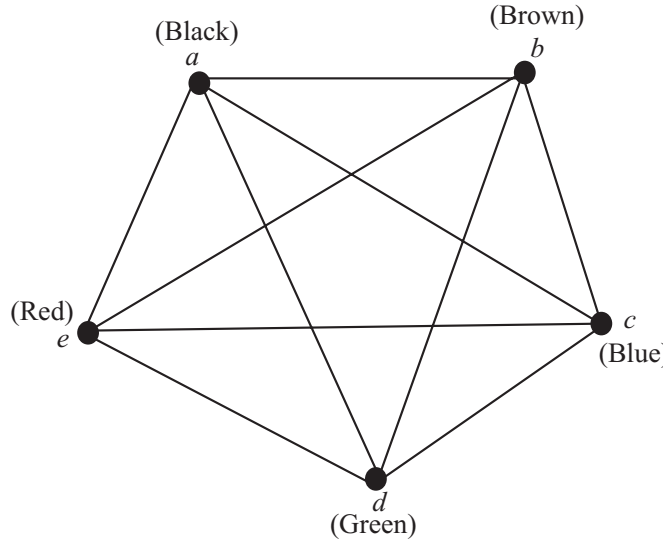


Fig. 15.8: A colouring of K_5 ■

Theorem 15.3: If G is k-critical, then $d \geq k - 1$.

Proof: This theorem can be proved by contradiction. If possible, let G be a k-critical graph with $\delta < k - 1$, and let v be a vertex of degree δ in G.

Since G is k-critical, $(G - v)$ is $(k - 1)$-colourable

Let $(V_1, V_2, V_3,...,V_{k-1})$ be a $(k–1)$-colouring of $(G - v)$.

By definition, v is adjacent us G to $\delta < (k - 1)$ vertices, and, therefore v must be nonadjacent in G to every vertex of some V_j. But then $(V_1, V_2, V_3...,V_j \cup \{v\},...,V_{k-1})$ is a $(k–1)$-colouring of G. (a contradiction).

Hence $\delta \geq k - 1$. ■

Theorem 15.4: In a critical graph, no vertex cut is a clique.

Proof: This result is to be proved by contradiction. Let G be a k-critical graph, and suppose that G has a vertex cut S that is a clique. We denote the S-component of G by $G_1, G_2, G_3,...,G_n$. Since G is k-critical, each G_i is $(k–1)$-colourable. Furthermore, because S is a clique, the vertices in S must receive distinct colours in any $(k–1)$-colouring of G_i. It follows that there are $(k–1)$-colourings of $G_1, G_2, G_3,...,G_n$ which agree on S.

But these colourings together yield a $(k–1)$-colouring of G, which is a contradiction. ■

Note: *For cliques refer Author's Graph Theory.*

Theorem 15.5: If G is simple, then $\pi_k(G) = \pi_k(G - e) — \pi_k(G \cdot e)$ for any edge e of G.

Proof: Let u and v be the ends of e. To each k-colouring of $(G - e)$ that assigns the same colour to u and v, there corresponds a k-colouring of $(G \cdot e)$ in which the vertex of $(G \cdot e)$ formed by identifying. u and v is assigned the common colour of u and v. This correspondence is clearly a bijection (refer Fig. 15.9). Therefore $\pi_k(G \cdot e)$ is precisely the number of k-colourings of $(G - e)$ in which u and v are assigned the same colour.

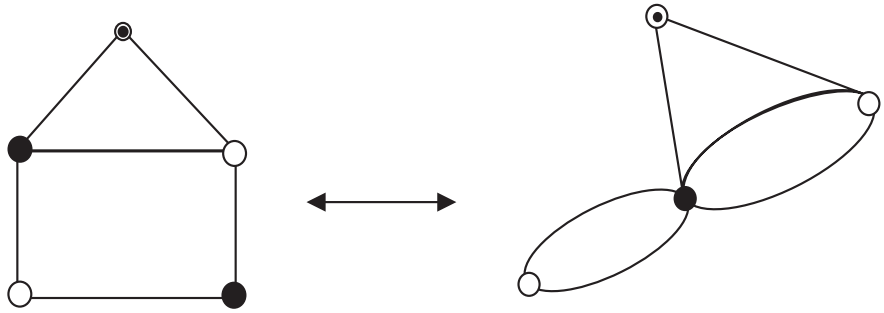


Fig. 15.9

Also, since each k-colouring of $(G - e)$ that assigns different colours to u and v is a k-colouring of G.

Conversely, $\pi_k(G)$ is the number of k-colourings of $(G - e)$ in which u and v are assigned different colours. It follows that $\pi_k(G - e) = \pi_k(G) + \pi_k(G \cdot e)$ ■

Example 15.5

Suppose that in one particular semester there are students taking each of the following combinations of courses.

- *Mathematics, English, Biology, Chemistry*
- *Mathematics, English, Computer Science, Geography*
- *Biology, Psychology, Geography, Spanish*
- *Biology, Computer Science, History, French*
- *English, Psychology, History, Computer Science*
- *Psychology, Chemistry, Computer Science, French*
- *Psychology, Geography, History, Spanish.*

What is the minimum number of examination periods required for exams in the ten courses specified so that students taking any of the given combinations of courses have no conflicts ?

Find a possible schedule which uses this minimum number of periods.

Solution

In order to picture the situation, we draw a graph with ten vertices labeled M, E, B, ... corresponding to Mathematics, English, Biology and so on, and join two vertices with an edge if exams in the corresponding subjects must not be scheduled together.

The minimum number of examination periods is evidently the chromatic number of this graph. What is this ? Since the graph contains K_5 (with vertices M, E, B, G, CS), at least five different colours are needed. (The exams in he subjects which these vertices represent must be scheduled at different times). Five colours are not enough, however, since P and H are adjacent to each other and to each of E, B, G and CS.

The chromatic number of the graph is, infact 6.

In Figure (15.10), we show a 6-colouring and the corresponding exam schedule.

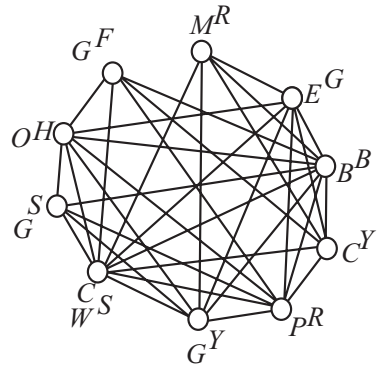


Fig. 15.10

Period 1	Mathematics, Psychology
Period 2	English, Spanish, French
Period 3	Biology
Period 4	Chemistry, Geography
Period 5	Computer Science
Period 6	History

■

Theorem 15.6: For any graph G, $\chi(G) \leq 1 + \max \delta(G')$,

Where the maximum is taken over all induced subgraph G' of G.

Proof: The result is obvious for totally disconnected graphs.

Let G be an arbitrary n-chromatic graph, $n \geq 2$.

Let H be any smallest induced subgraph such that $\chi(H) = n$

The graph H therefore has the property that

$\chi(H - v) = n - 1$ for all its points v.

It follows that deg $v \geq n - 1$ so that $\delta(H) \geq n - 1$ and hence

$$n - 1 \leq \delta(H) \leq \max \delta(H') \leq \max \delta(G')$$

The first maximum taken over all induced subgraphs H' of H and the second over all induced subgraphs G' of G.

This implies that

$$\chi(G) = n < 1 + \max \delta(G')$$

Corollary: For any graph G, the chromatic number is atmost one greater than the maximum degree $\chi \leq 1 + \Delta$.

Theorem 15.7: For any graph G, the sum and product of χ and χ satisfy the inequalities:

$$2\sqrt{P} \leq \chi + \bar{\chi} \leq P + 1.$$

$$P \leq \chi\,\bar{\chi} \leq \left(\frac{P+1}{2}\right)^2$$

Proof: Let G be n-chromatic and let $V_1, V_2, \ldots V_m$ be the colour classes of G, where $|V_i| = P_i$

Then of course $\sum P_i = P$ and $\max P_i \geq \frac{P}{n}$.

Since each V_i induces a complete subgraph of $\bar{G}$

$$\bar{\chi} \geq \max P_i \geq \frac{P}{n} \text{ so that } \chi\,\bar{\chi} \geq 2\sqrt{P}.$$

This establishes both lower bounds.

To show that $\chi + \bar{\chi} \leq P + 1$, we use induction on P, noting that equality holds when $P = 1$.

We thus assume that $\chi(G) + \chi(G) \leq P$ for all graphs G having $P - 1$ points.

Let H and $\bar{H}$ be complementary graphs with P points, and let v be a point of H.

Let H and $\bar{H}$ be complementary graphs with P points, and let v be a point of H.

Then $G = H - v$ and $\bar{G} + \bar{H} - v$ are complementary graphs with $P - 1$ points.

Let the degree of v in H be d so that the degree of v in $\bar{H}$ is $P - d - 1$. It is obvious that

$$\chi(H) \leq \chi(G) + 1 \text{ and } \bar{\chi}(H) \leq \chi(G) + 1$$

If either

$$\chi(H) < \chi(G) + 1 \text{ or } \bar{\chi}(H) < \bar{\chi}(G) + 1.$$

then $\chi(H) + \bar{\chi}(H) \leq P + 1$.

Suppose then that $\chi(H) = \chi(G) + 1$ and $\bar{\chi}(H) = \bar{\chi}(G) + 1$.

This implies that the removal of v from H, producing G, decreases the chromatic number so that $d \geq \chi(G)$.

Similarly $P - d - 1 \geq \overline{\chi}(G)$,

thus $\chi(G) + \overline{\chi}(G) \leq P - 1$

Therefore, we always have $\chi(H) + \overline{\chi}(H) \leq P + 1$

Finally, applying the inequality

$4\chi\overline{\chi} \leq (\chi + \overline{\chi})^2$ we see that $\chi\overline{\chi} \leq \left[\frac{(P+1)}{2}\right]^2$ ■

Theorem 15.8: If G is a graph with n vertices and degree δ then $\chi(G) \geq \frac{n}{n-\delta}$.

Proof: Recall that δ is the minimum of the degrees of vertices.

Therefore, every vertex v of G has atleast δ number of vertices adjacent to it.

Hence there are at most $n - \delta$ vertices can have the same colour.

Let K be the least number of colours with which G can be properly coloured. Then $K = \chi(G)$.

Let $\alpha_1, \alpha_2, ..., \alpha_K$ be these colours and let n_1 be the number of vertices having colour α_1, n_2 be the number of vertices having colour α_2 and so on, and finally n_K be the number of vertices having colour α_K.

Then $n_1 + n_2 + n_3 + ... \quad n_k = n$ ...(1)

and $n_1 \leq n - \delta, n_2 \leq n - \delta, ... n_k \leq n - \delta$ ...(2)

Adding the K in equalities in (2), we obtain

$$n_1 + n_2 + ... n_k \leq K(n - d)$$

or $n \leq K(n - d)$, using (1)

Since $K = \chi(G)$, this becomes

$$\chi(G) \geq \frac{n}{n-\delta}$$

This is required result. ■

15.5 Edge Colouring

A k-edge colouring of a loopless graph G is an assignment of k colours, 1, 2, 3, ..., k, to the edges of G. The colouring is proper if no two adjacent edges have the same colour.

Alternatively, a k-edge colouring can be thought of as a partition $(E_1, E_2, E_3...,E_k)$ of E where E_i denotes the possibly empty subset of E assigned colour i. A proper k-edge colouring is than a k-edge colouring $(E_1, E_2, E_3...,E_k)$ in which each subset E_i is a matching.

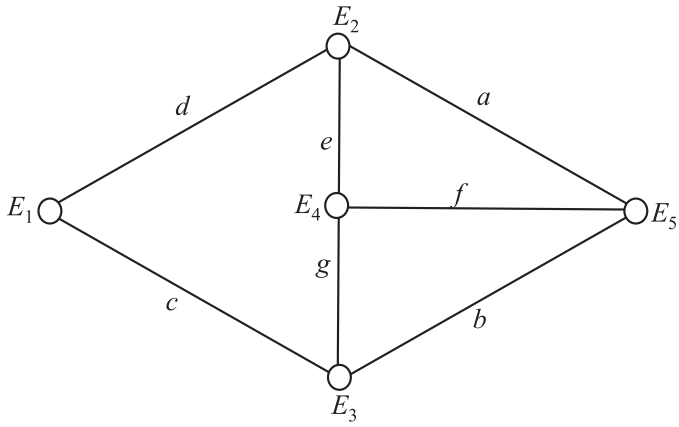


Fig. 15.11: Edge Colouring

The proper four edge colouring for Fig. 15.11 is $(\{a, g\}, \{b, e\}, \{c,f\}, \{d\})$

G is k-edge-colourable if G has a proper k-edge colouring.

Trivially, every loopless graph G is E-edge colourable.

If G is k-edge-colourable, then G is also l-edge-colourable for every $1 > k$. The edge chromatic number $\chi'(G)$, of a loopless graph G, is the minimum k for which G is k-edgecolourable. G is k-edge-chromatic if $\chi'(G) = k$. It can be readily verified that the graph of Fig.15.11 has no proper 3-edge colouring. This graph is therefore 4-edge-chromatic.

Clearly, in any proper edge colouring, the edges incident with any one vertex must be assigned different colours. It follows that

$$\chi' \geq \Delta \qquad ...(i)$$

We shall show that, in the case when G is bipertite,

$\chi' = \Delta$. we can say that colour i is represented at vertex v if same edge incident with v has colour i.

Lemma 15.1: Let G be a connected graph that is not an odd cycle. Then G has a 2-edge colouring in which both colours are represented at each vertex of degree at least two.

Proof: We may clearly assume that G is nontrivial. Suppose, first, that G is eulerian. If G is an even cycle, the proper 2-edge colouring of G has the

required property. Otherwise, G has a vertex v_0 of degree at least four. Let $v_o e_1 v_1 ... e. v_0$ be an Euler tour of G^* and set

$$E_1 = \{e, : i \text{ odd}\}$$

and $$E_2 = \{e_i : i \text{ even}\} \quad ...(i)$$

Then the 2-edge colouring $(E,, E_2)$ of G has the required property, since each vertex of G is an internal vertex of $v_0 e_1 v_1 ... e_{\in}, v_0$.

If G is not eulerian, construct a new graph G^* by adding a new vertex v_0 and joining it to each vertex of odd degree in G. Clearly G^* is eulerian. Let $v_0 e_1, v_1, ... e_f v_0$ be an Euler tour of G^* and define E_1 and E_2 as in (is then easily verified that the 2-edge colouring (E_1, Ç E, E_2 Ç E) of G has the required property. ■

Lemma 15.2: Let $\mathscr{C} = (E_1, E_2, ... , E_k)$ be an optimal k-edge colouring of G. If there is a vertex u, in G and, colours i and j such that i is not represented at u and j is represented at least twice at u, then the component of $G[E_i, \cup E_j]$ that contains u is an odd cycle.

Proof: Let u be a vertex that satisfies the hypothesis of the lemma, and denote by H the component of $G[E_i \ U \ E_j]$ containing u. Suppose that H is not an odd cycle. Then, by lemma 15.1, H has a 2-edge colouring in which both colours are represented at each vertex of degree at least two in H. When we recolour the edges of H with colours i and j in this way, we obtain a new k-edge colouring $\mathscr{C}$.

$b' = (E'_1, E'_2, ... , E'_k)$ of G. Denoting by $c'(v)$ the number of distinct colours at v in the colouring $\mathscr{C}$.

we have

$$c'(u) = c(u) + 1$$

since, now, both i and j are represented at u, and also

$$c'(v) \geq c(v) \text{ for } v \neq u$$

Thus $\sum_{v \in V} c'(v) > \sum_{v \in V} c(v)$, contradicting the choice of $\mathscr{C}$.

It follows that H is indeed an odd cycle ■

Theorem 15.9: ***(König's Theorem) (1916)***

Every bipartite graph G satisfies $\chi'(G) = \Delta(G)$.

Proof: We apply induction on $||G||$. For $||G|| = 0$ the assertion holds. Now assume that $||G|| \geq 1$, and that the assertion holds for graphs with fewer edges. Let $\Delta := \Delta(G)$, pick an edge $xy \in G$, and choose a Δedge-colouring

of $(G - xy)$ by the induction hypothesis. Let us refer to the edges coloured α as α-edges, etc.

In $(G - xy)$ each of x and y is incident with at most $(\Delta - 1)$ edges. Hence there are $\alpha, \beta \in \{1, \ldots, \Delta)$ such that x is not incident with an α-edge and y is not incident with a (β-edge. If $\alpha = \beta$, we can colour the edge xy with this colour and are done; so we may assume that $\alpha \neq \beta$, and that x is incident with a β-edge.

Let us extend this edge to a maximal walk W whose edges are coloured b and a alternately. Since no such walk contains a vertex twice (why not?), W exists and is a path. Moreover, W does not contain y: if it did, it would end in y on an a-edge (by the choice of b) and thus have even length, so $(W + xy)$ would be an odd cycle in G. We now recolour all the edges on W, swapping a with b. By the choice of a and the maximality of W, adjacent edges $(G - xy)$ are still coloured differently. We have thus found a D-edge-colouring of $(G - xy)$ in which neither x nor y is incident with a b-edge. Colouring xy with b, we extend this colouring to a D-edge-colouring of G.

Theorem 15.10: ***(Vizing Theorem, 1964)***

Every graph G satisfies

$$\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1.$$

Proof: We prove the second inequality by induction on $\|G\|$. For $\|G\| = 0$ it is trivial. For the induction step let $G = (V, E)$ with $\Delta := \Delta(G) > 0$ be given, and assume that the assertion holds for graphs with fewer edges. Instead of '$(\Delta + 1)$-edge-colouring' let us just say '*colouring*'. An edge coloured α will again be called an α-*edge*.

For every edge $e \in G$ there exists a colouring of $(G - e)$ by the induction hypothesis. In such a colouring, the edges at a given vertex v use at most $d(v) \leq \Delta$ colours, so some colour $\beta \in \{1, \ldots, \Delta + 1\}$ is missing at v. For any other colour a, there is a unique maximal walk (possibly trivial) starting at v, whose edges are coloured alternately α and β. This walk is a path; we call it the α/β - path from v.

Suppose that G has no colouring. Then the following holds:

Given $xy \in E$, and any colouring of $(G - xy)$ in which the colour α is missing at x and the colour β is missing at y,(1) the α/β-path from y ends in x.

Otherwise we could interchange the colours α and β along this path and colour xy with α, obtaining a colouring of G (contradiction).

Let $xy_0 \in G$ be an edge. By induction, $G_0 := G - xy_0$ has a colouring c_0. Let α be a colour missing at x in this colouring. Further, let $y_0, y_1, \ldots y_k$ be a maximal sequence of distinct neighbours of x in G, such that $c_0(xy_i)$ is missing in c_0 at y_{i-1} for each $i = 1, \ldots, k$. For each of the graphs $G_i := G - xy_i$ we define a colouring c_i, setting

$$c_i(e) := \begin{cases} c_0(xy_{j+1}) & \text{for } e = xy_j \text{ with } j \in \{0,\ldots,i-1\} \\ c_0(e) & \text{otherwise;} \end{cases}$$

note that in each of these colourings the same colours are missing at x as in c_0.

Now let β be a colour missing at y_k in c_0. Clearly, β is still missing at y_k in c_k. If β were also missing at x, we could colour xy_k with β and thus extend c_k to a colouring of G. Hence, x is incident with a β-edge (in every colouring). By the maximality of k, therefore, there is an $i \in \{1,\ldots,k-1\}$ such that $c_0(xy_i) = b$.

Let P be the α/β-path from yk in G_k (with respect to c_k; Fig. 15.12). By (1), P ends in x, and it does so on a β-edge, since α is missing at x. As $\beta = c_0(xy_i) = c_k(xy_{i-1})$, this is the edge xy_{i-1}. In c_0, however, and hence also in c_{i-1}, β is missing at y_{i-1} (by (2) and the choice of y_i); let P' be the α/β-path from y_{i-1} in G_{i-1} (with respect to c_{i-1}). Since P' is uniquely determined, it starts with $y_{i-1}Py_k$; note that the edges of P_x are coloured the same in c_{i-1} as in c_k. But in c_0, and hence in c_{i-1}, there is no β-edge at y_k (by the choice of β). Therefore P' ends in y_k, contradicting (1).

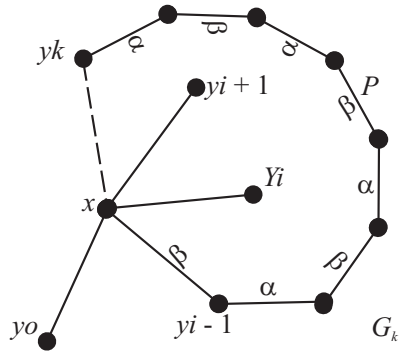


Fig. 15.12: The α /β -path P in G_k

Note: Vizing's theorem divides the finite graphs into two classes according to their chromatic index; graphs satisfying $\chi' = \Delta$ are called (imaginatively) class 1, those with $\chi' = \Delta + 1$ are class 2.

15.6 List Colouring

Suppose we are given a graph $G = (V, E)$, and for each vertex of G a list of colours permitted at that particular vertex: when can we colour G (in the usual sense) so that each vertex receives a colour from its list? More formally, let $(S_v)_{v\in V}$ be a family of sets. We call a vertex colouring c of G with $c(v) \in S_v$, for all $v \in V$ a colouring from the lists S_v. The graph G is called k-list-colourable, or k-choosable, if, for every family $(S_v)_{v\in V}$ with $|S_v| = k$ for all v, there is a vertex colouring of G from the lists S_v. The least integer k for which G is k-choosable is the list-chromatic number, or choice number $\mathrm{ch}(G)$ of G.

List-colourings of edges are defined analogously. The least integer k such that G has an edge colouring from any family of lists of size k is the list-chromatic index $\mathrm{ch}'(G)$ of G; formally, we just set $\mathrm{ch}'(G) := \mathrm{ch}(L(G))$, where $L(G)$ is the line graph of G.

In principle, showing that a given graph is k-choosable is more difficult than proving it to be k-colourable: the latter is just the special case of the former where all lists are equal to $\{1, ..., k\}$. Thus,

$$\mathrm{ch}(G) \geq \chi(G) \text{ and } \mathrm{ch}'(G) \geq \chi'(G)$$

for all graphs G.

List colouring conjecture. Every graph G satisfies $\mathrm{ch}'(G) = \chi'(G)$.

We shall prove the list colouring conjecture for bipartite graphs. As a tool we shall use orientations of graphs. If D is a directed graph and $v \in V(D)$, we denote by $N^+(v)$ the set, and by $d^+(v)$ the number, of vertices w such that D contains an edge directed from v to w.

To see how orientations come into play in the context of colouring, let us recall the greedy algorithm. In order to apply the algorithm to a graph G, we first have to choose a vertex enumeration $v_1, ..., v_n$ of G. The enumeration chosen defines an orientation of G: just orient every edge v_iv_j 'backwards', from v_i to v_j if $i > j$. Then, for each vertex v_i to be coloured, the algorithm considers only those edges at v_i that are directed away from v_i: if $d^+(v) < k$ for all vertices v, it will use at most k colours. Moreover, the first colour class U found by the algorithm has the following property: it is an independent set of vertices to which every other vertex sends an edge. The second colour class has the same property in $G - U$, and so on.

The following lemma generalizes this to orientations D of G that do not necessarily come from a vertex enumeration, but may contain some if, for every vertex $v \in D - U$, there is an edge in D directed from

v to a vertex in U. Note that kernels of non-empty directed graphs are themselves non-empty.

Lemma 15.3: Let H be a graph and $(S_v)_{v \in V(H)}$ a family of lists. If H has an orientation D with $d^+(v) < |S_v|$ for every v, and such that every induced subgraph of D has a kernel, then H can be coloured from the lists S_v.

Proof: We apply induction on $|H|$. For $|H| = 0$ we take the empty colouring. For the induction step, let-$|H| > 0$. Let α be a colour occurring in one of the lists S_v and let D be an orientation of H as stated. The vertices v with $\alpha \in S_v$ span a non-empty subgraph D' in D; by assumption, D' has a kernel $U \neq \theta$.

Let us colour the vertices in U with α, and remove α from the lists of all the other vertices of D'. Since each of those vertices sends an edge to U, the modified lists S'_v for $v \in D - U$ again satisfy the condition $d^+(v) < |S'_v|$ in $D - U$. Since $D - U$ is an orientation of $H - U$, we can thus colour $H - U$ from those lists by the induction hypothesis. As none of these lists contains α, this extends our colouring $U \to \{\alpha\}$ to the desired list colouring of H. ■

15.7 Greedy Colouring

The greedy algorithm considers the vertices in a specific order $V_1,...,V_n$ and assigns to V_i the smallest available color not used by v_i's neighbours among $V_1,...,V_{i-1}$, adding a fresh color if needed. The quality of the resulting coloring depends on the chosen ordering. There exists an ordering that leads to a greedy coloring with the optimal number of $\chi(G)$ colors. On the other hand, greedy colorings can be arbitrarily bad; for example, the crown graph on n vertices can be 2-colored, but has an ordering that leads to a greedy coloring with $n / 2$ colors.

If the vertices are ordered according to their degrees, the resulting greedy coloring uses at most max_i min $\{d(x_i) + 1, i\}$ colors, at most one more than the graph's maximum degree. This heuristic is sometimes called the Welsh–Powell algorithm. Another heuristic establishes the ordering dynamically while the algorithm proceeds, choosing next the vertex adjacent to the largest number of different colors. Many other graph coloring heuristics are similarly based on greedy coloring for a specific static or dynamic strategy of ordering the vertices, these algorithms are sometimes called *sequential coloring algorithms.*

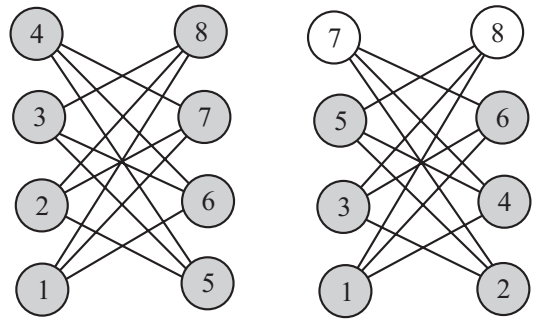


Fig. 15.13: Two Greedy Colorings of the same Graph using Different Vertex Orders. The Right Example Generalises to 2-colourable Graphs with *n* Vertices when the Greedy Algorithm Expends *n*/2 Colours

Theorem 15.11: ***Four Colour Problem***

Every planar graph has a chromatic number of four or less.

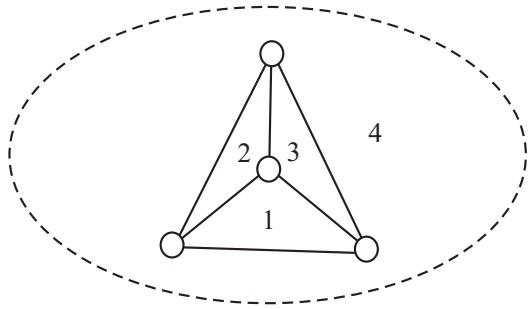


Fig. 15.14: Four Colour Problem

Proof: We assume the four colour conjecture holds and let G be any plane map.

Let G^* be the underlying graph of the geometric dual of G.

Since two regions of G are adjacent if and only if the corresponding vertices of G^* are adjacent, map G is 4-colorable because graph G^* is 4-colorable.

Conversely, assume that every plane map is 4-colorable and let H be any planar graph.

Without loss of generality, we suppose H is a connected plane graph.

Let H^* be the dual of H, so drawn that each region of H^* encloses precisely one vertex of H. The connected plane pseudograph H^* can be converted into a plane graph H' by introducing two vertices into each loop of H^* and adding a new vertex into each edge in a set of multiple edges.

The 4-colorability of H' now implies that H is 4-colorable, completing the verification of the equivalence.

If the four color conjecture is ever proved, the result will be best possible, for it is easy to give examples of planar graphs which are 4-chromatic, such as K_4 and W_6.

Theorem 15.12: ***Five Colour Theorem***

Every planar graph is 5-colourable.

Proof: Let G be a plane graph with $n \geq 6$ vertices and m edges. We assume inductively that every plane graph with fewer than n vertices can be 5-coloured. We have,

$$d(G) = 2m/n \leq 2(3n - 6)/n < 6 ;$$

let $v \in G$ be a vertex of degree at most 5. By the induction hypothesis, the graph $H := G - v$ has a vertex colouring c: $V(H) \to (1,...,5\ \}$. If c uses at most 4 colours for the neighbours of v, we can extend it to a 5colouring of G. Let us assume, therefore, that v has exactly 5 neighbours, and that these have distinct colours.

Let D be an open disc around v, so small that it meets only those five straight edge segments of G that contain v. Let us enumerate these segments according to their cyclic position in D as s_1, ... , s_5, and let vv_i be the edge containing s_i (i = 1, ... , 5; Fig. 15.15) Without loss of generality we may assume that $c(v_i) = i$ for each i.

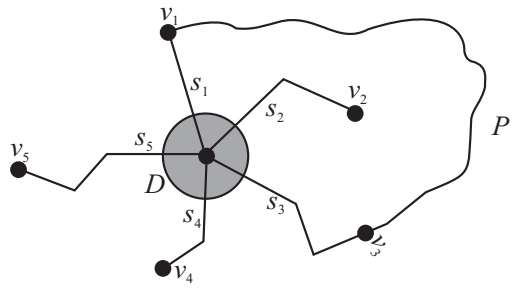


Fig. 15.15: The Proof of the Five Colour Theorem

Let us show first that every v_1–v_3 path $P \subseteq H$ separates v_2 from v_4 in H. Clearly, this is the case if and only if the cycle $C := vv_jPv_3v$ separates v_2 from v_4 in G. We prove this by showing that v_2 and v_4 lie in different faces of C.

Consider the two regions of $D \setminus (s_1 \cup s_3)$. One of these regions meets s_2, the other s_4. Since $C \cap D \subseteq s_1 \cup s_3$, the two regions are each contained

within a face of C. Moreover, these faces are distinct: otherwise, D would meet only one face of C, contrary to the fact that v lies on the boundary of both faces. Thus $D \cap s_2$ and $D \cap s_4$ lie in distinct faces of C. As C meets the edges $vv_2 \supseteq s_2$ and $vv_4 \supseteq s_4$ only in v, the same holds for v_2 and v_4.

Given $i, j \in \{1, \ldots, 5\}$, let $H_{i,j}$ be the subgraph of H induced by the vertices coloured i or j. We may assume that the component C_1 of $H_{1,3}$ containing v_1 also contains v_3. Indeed, if we interchange the colours 1 and 3 at all the vertices of C_1, we obtain another 5-colouring of H; if $v_3 \notin C_1$, then v_1 and v_3 are both coloured 3 in this new colouring, and we may assign colour 1 to v. Thus, $H_{1,3}$ contains a $v_1 - v_3$ path P. As shown above, P separates v_2 from v_4 in H. Since $P \cap H_{2,4} = 0$, this means that v_2 and v_4 lie in different components of $H_{2,4}$. In the component containing v_2, we now interchange the colours 2 and 4, thus recolouring v_2 with colour 4. Now v no longer has a neighbour coloured 2, and we may give it this colour.

Example 15.6

Find all possible maximal independent sets of the following graph using Boolean expression.

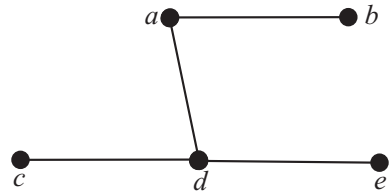


Fig. 15.16

Solution

The Boolean expression for this graph

$$\phi = \Sigma xy = ab + ad + cd + de \text{ and}$$

$$\begin{aligned}
\phi' &= (a' + b')(a' + b')(d' + e') \\
&= \{a'(a' + d') + b'(a' + d')\} \\
&\quad \{c'(d' + e') + (d'(d' + e')\} \\
&= \{a' + b'a' + b'd'\}\ \{c'd' + c'e' + d'\} \\
&= \{a'(1 + b') + b'd'\}\{d'(c' + 1) + (c'e'\} \\
&= \{a' + b'd'\}\{d' + c'e'\}
\end{aligned}$$

$$= a'd' + a'c'e' + b'd' + b'c'd'e'$$
$$= a'd' + a'c'e' + b'd' (1 + c'e')$$
$$= a'd' + a'c'e' + b'd'$$

Thus, $\phi_1 = a'd', f_2 = a'c'e'$

and $\phi_3 = b'd'$.

Hence maximal independent sets are

$$V - \{a, b\} = \{b, c, e\}$$
$$V - \{a, c, e\} + \{b, d\}$$

and $V - \{b, d\} = \{a, c, e\}$. ■

Example 15.7

Prove that for a graph G with n vertices

$$\beta(G) \geq \frac{n}{\chi(G)}.$$

Solution

Let K be the minimum number of colours with which G can be properly colored.

Then $K = \chi(G)$. Let $a_1, a_2, ..., a_K$ be these colours and let $n_1, n_2, ... n_K$ be the number of vertices having colours $\alpha_1, \alpha_2, ..., \alpha_K$ respectively.

Then $n_1, n_2, ..., n_K$ are the orders of the maximal independent sets, because a set of all vertices having the same colour contain all vertices which are mutually non-adjacent.

Since $\beta(G)$ is the order of a maximal independent set with largest number of vertices, none of $n_1, n_2, ..., n_K$ can exceed $\beta(G)$.

i.e., $n_1 \leq \beta(G), n_2 \leq \beta(G), ..., n_K \leq \beta(G)$ ■

15.8 Applications

15.8.1 The Time Table Problem

In a school, there are m teachers $X_1, X_2, ..., X_m$, and n classes $Y_1, Y_2, ..., Y_n$. Given that teacher X_i is required to teach class Y_i for p_{ij} periods, schedule a complete timetable in the minimum possible number of periods.

The above problem is known as the timetabling problem, and can be solved completely using the theory of edge colourings developed in this chapter. We represent the teaching requirements by a bipartite graph G with bipartition (X, Y), where $X = \{x_1, x_2, \ldots, x_m\}$, $Y = \{y_1, y_2, \ldots, y_n\}$ and vertices x_i and y_i are joined by p_{ij} edges. Now, in any one period, each teacher can teach at most one class, and each class can be taught by at most one teacher—this, at least, is our assumption. Thus a teaching schedule for one period corresponds to a matching in the graph and, conversely, each matching corresponds to a possible assignment of teachers to classes for one period. Our problem, therefore, is, to partition the edges of G into as few matchings as possible or, equivalently, to properly colour the edges of G with as few colours as possible. Since G is bipartite. Hence, if no teacher teaches for more than p periods, and if no class is taught for more than p periods, the teaching requirements can be scheduled in a p-period timetable. Furthermore, there is a good algorithm for constructing such a timetable. We thus have a complete solution to the timetabling problem.

However, the situation might not be so straightforward. Let us assume that only a limited number of classrooms are available. With this additional constraint, how many periods are now needed to schedule a complete timetable?

Suppose that altogether there are l lessons to be given, and that they have been scheduled in a p-period timetable. Since this timetable requires an average of l/p lessons to be given per period, it is clear that at least $\{l/p\}$ rooms will be needed in some one period. It turns out that one can always arrange l lessons in a p-period timetable so that at most $\{l/p\}$ rooms are occupied in any one period.

15.8.2 Scheduling of Jobs

Vertex coloring models to a number of scheduling problems. In the cleanest form, a given set of jobs need to be assigned to time slots, each job requires one such slot. Jobs can be scheduled in any order, but pairs of jobs may be in conflict in the sense that they may not be assigned to the same tine slot. For example because they both rely on a shared resource. The corresponding graph contains a vertex for every job and an edge for every conflicting pair of jobs. The chromatic number of the graph is exactly the minimum makes an, the optimal time to finish all jobs without conflicts.

Details of the scheduling problem define the structure of the graph. For example, when assigning aircrafts to flights, the resulting conflict graph is an interval graph, so the coloring problem can be solved efficiently. In bandwidth allocation to radio stations, the resulting conflict graph is a unit disk graph, so the coloring problem is 3-approximable.

15.8.3 Ramsey Theory

An important class of improper coloring problems is studied in Ramsey theory, where the graph's edges are assigned to colors, and there is no restriction on the colors of incident edges. A simple example is the friendship theorem says that in any coloring of the edges of K_6 the complete graph of six vertices there will be a monochromatic triangle; often illustrated by saying that any group of six people either has three mutual strangers or three mutual acquaintances. Ramsey theory is concerned with generalisations of this idea to seek regularity amid disorder, finding general conditions for the existence of monochromatic subgraphs with given structure.

15.8.4 Storage Problem

A company manufactures n chemicals $C_1, C_2, ..., C_n$. Certain pairs of these chemicals are incompatible and would cause explosions if brought into contact with each other. As a precautionary measure the company wishes to partition its warehouse into compartments, and store incompatible chemicals in different compartments. What is the least number of compartments into which the warehouse should be partitioned?

We obtain a graph G on the vertex set $\{v_1, v_2, ..., v_n\}$ by joining two vertices v_i and v_j if and only if the chemicals C_i and C_j are incompatible. It is easy to see that the least number of compartments into which the warehouse should be partitioned is equal to the chromatic number of G.

The solution of many problems of practical interest (of which the storage problem is one instance) involves finding the chromatic number of a graph. Unfortunately, no good algorithm is known for determining the chromatic number. Here we describe a systematic procedure which is basically 'enumerative' in nature. It is not very efficient for large graphs.

Since the chromatic number of a graph is the least'number of independent sets into which its vertex set can be partitioned, we begin by describing a method for listing all the independent sets in a graph. Because every independent set is a subset of a maximal independent set,

it suffices to determine all the maximal independent sets. In fact, our procedure first determines complements of maximal independent sets, that is, minimal coverings.

SUMMARY

1. A graph in which every vertex has been assigned a colour according to a proper colouring is called a properly coloured graph.
2. A colouring using at most k colour is called a proper k-colouring.
3. A Chromato Polynomial includes at least as much information about the colourability of G as does the Chromatic number.
4. Every Chromatic number of any bipartite graph is always 2.
5. A minimum number of hours for a scheduled of committee meeting in one scheduling problem is $X(G_0)$.
6. If G in k-critically, then $d \geq k$-1.
7. In a critical graph, no vertex cut is a clique.
8. A k-edge colouring of a loopless graph G is an assignment of k colours 1, 2, 3, ..., k, to the edge of G.
9. The colouring is proper if no two adjacent edges have the same colour.
10. Every planar graph is 5-colourable.

EXERCISE

1. Describe a good algorithm for finding a proper $(\Delta + 1)$-edge colouring of a simple graph G.
2. Show that if G is simple with d > 1, then G has a $(d - 1)$-edge colouring such that all $(d - 1)$ colours are represented at each vertex.
3. If G is bipertite, then prove that $c' = D$.
4. If G is simple, Hence prove that either $c' = D$ or $c' = \Delta + 1$.
5. The product of simple graphs G and H is the simple graph $G \times H$ with vertex set $V(G) \times V(H)$, in which (u, v) is adjacent to (u', v') if and only if either $u = u'$ and $vv' \in E(H)$ or $v = v'$ and $uu' \in E(G)$.
   (*a*) Using Vizing's theorem, show that $c'(G \times K_2) = D(G \times K_2)$.
   (*b*) Deduce that if H is nontrivial with $c'(H) = D(H)$, then $c'(G \times H) = D(G \times H)$.

6. In a school there are seven teachers and twelve classes. The teaching requirements for a five-day week are given by the matrix

	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	Y_{10}	Y_{11}	Y_{12}
X_1	3	2	3	3	3	3	3	3	3	3	3	3
X_2	1	3	6	0	4	2	5	1	3	3	0	4
X_3	5	0	5	5	0	0	5	0	5	0	5	5
X_4	2	4	2	4	2	4	2	4	2	4	2	3
X_5	3	5	2	2	0	3	1	4	4	3	2	5
$X6$	0	3	4	3	4	3	4	3	4	3	3	0

where p_{ij} is the number of periods that teacher X_i must teach class Y_j.

(*a*) Into how many periods must a day be divided so that requirements can be satisfied?

(*b*) If an eight-period/day timetable is drawn up, how many classrooms will be needed?

Suggested Readings

1. R.P. Grimaldi, *Discrete and Combinatorial Mathematics*: *An Applied Introduction,* 4th ed., Addison-Wesley, Reading, MA, 1999 pp. 547-590.

2. E. Horowitz and S. Sahni, *Fundamentals of Computer Algorithms,* Computer Science Press, Rockville, MD, 1978, pp. 152-370.

3. A.Tucker, *Applied Combinatorics*, 2nd ed., Wiley, New York, NY, 1984, pp. 80-122.

4. R.J.Wilson and J.J. Watkins, *Graphs*: *An Introductory Approach*, Wiley, New York, 1990, pp. 185-214.

❑❑❑

References

1. Agnorsson G, Greenlaw R, *Graph Theory: Modeling, Applications and Algorithms,* Pearson Education, 2012.
2. M.O. Albertson, J.P. Hutchinson, *Discrete Mathematics with Algorithms* Wiley, 2000.
3. Alan Tucher, *Applied Combinatories,* Wiley, 2002.
4. Arumugam S, Ramachandran S, *Invitations to Graph Theory,* Scitech Publications, 2006.
5. Balakrishanan V.K. *Theory and problems of Graph Theory,* TMH, 2008.
6. G.S.S. BhishmaRao, *Discrete Structure and Graph Theory,* Scitech, 2003.
7. Bollobas B, *Modern Graph Theory*, Springer, 2003.
8. Bondy and Murty, *Graph Theory with applications,* Macmillan, 2001.
9. Bore N.K. and Hiang P, *Neural Networks Fundamentals with Graphs,* Algorithms and Applications, TMH, 2006.
10. Chandrashekhariah, *Discrete Mathematical Structures,* prism, 2000.
11. Chandrashekharan N, Umaparvathi M, *Discrete Mathematics,* PHI, 2010.
12. Deo Narsingh, *Graph Theory with Application to Engineering and computer science,* PHI, 2010.
13. Diestal R, *Graph Theory*, Springer, 2003.
14. Edward and Bender, *Basic concepts in Graph Theory,* EWS publication USA, 1999.
15. Godsil and Rogle, *Algebraic Graph Theory* springer, 2002.
16. Gromaldi, Ramana, Discrete and combinational Mathematics, an applied introduction, pearson education, 2011.

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2

17. Goodaria E.G., Parmenter M.M, *Discrete Mathematics with Graph Theory,* PHI, 2006.

18. Harary. F, *Graph Theory,* Narosa, 2006.

19. Hoperoft J.E., Dillman J.U., *Introduction to Automata Theory, languages and computation,* Narosa, 2001.

20. Kolman, Busby, Ross, *Discrete Mathematical Structures,* Person, 2010.

21. Lipschutz.S, *Discrete Mathematics (Schaum's series),* TMH, 2006.

22. Liu C.L., Mohapatra D.P., *Elements of Discrete Mathematics, a computer oriental approach,* TMH-2011.

23. Mott, Kandel, Baker, *Discrete Mathematics for computer scientists and Mathematicians,* PHI, 2001.

24. Pelikan J, Vesztergombi, L. Lovasz, *Discrete Mathematics Elementary and Beyond,* Springer, 2006.

25. Rakesh Dubey, Adesh Pandey, Ritu Gupta, *Discrete structure and Automata theory,* Cambridge, 2010.

26. Rosen K.H., *Discrete Mathematics and Its Applications with combinatorics and Graph Theory,* TMH, 2008.

27. Rudraiah N, Shivakumar B.S. *Discrete Mathematical Structures,* Sapna, 2001.

28. Grimani, P.K., Nasir S.F.B., *A textbook on Automata Theory* Cambridge, 2008.

29. Thomas Koshy, *Discrete Mathematics with Applications,* Elsevier, Academic Press, 2006.

30. Tremblay, Manohar, *Discrete Mathematics Structure with application to computer science,* TMH-2006.

31. Veerrajan T, *Discrete Mathematics with Graph Theory and combinatorics,* TMH, 2012.

32. Wallis, W.D., *A biginnerr Guide to discrete mathematics Birkhauser, 2012.*

33. Yadav S.K., *Discrete Mathematical Structures,* Satya prakashan, 2001.

34. Yadav S.K., *Element of Graph Theory,* Ane books, 2011.

Index

A

Absolute value, 247
Abstract machine, 396
Accepting states, 414
Acyclic graph, 554
Addition of two graphs, 456
Adjacency matrix of a relation, 198
Airline connections, 472
Alarm buzzing, 414
Algebraic terms and operations, 453
Alternating turing machine, 395
Ambiguous grammar, 411
Angle brackets, 404
Antecedent rules, 154
Applications of graphs, 471
Attraction, 472
Automatic proving system, 152
Axiom scheme, 153

B

Backus naur form, 407
Backus-normal form (BNF), 407
Basis clause, 319
Belief and plausibility measures, 373
Biconnected graph, 452
Bijective homomorphism, 309
Bijective mapping, 243
Binary relations in a set, 183
Binary search tree, 518
Binary trees, 492
Bipertite graph, 524
Boolean isomorphism, 309
Boolean matrices, 194
Boolean power, 195
Boolean product, 195
Bounded lattice, 290
Bounds of the chromatic number, 614
Breadth – first search, 496
BSF algorithm, 496

C

Cantor's diagonalization procedure, 258
Ceiling function, 245
Cell probe model, 395
Cellular automation, 394
Characteristic function of a set, 259
Child, 485
Chomsky grammar, 410
Chromatic polynomial, 613
Circuit and cycle, 447
Closure of relations, 207
Codomain, 239, 448
Colouring conjecture, 630
Combinatorial circuits, 326
Complement of a fuzzy set, 362
Complete lattice, 290
Complete matching, 590
Composition, 371
Composition of functions, 248
Composition of relations, 208
Computer network problem, 432
Concatenation of languages, 398
Conjunctive normal form, 145
Connected graph, 449, 494
Connectives, 152
Consequent rules, 154
Constant mapping, 243
Containment of fuzzy sets, 361

© The Author(s) 2023

S. K. Yadav, *Discrete Mathematics with Graph Theory*,
https://doi.org/10.1007/978-3-031-21321-2

Context-free and regular grammars, 410
Context-sensitive, 410
Convex sublattice, 291
Corresponding digraph, 444
Countable and uncountable sets, 257
Counting to five, 414
Covering, 582
Covering to switching functions, 602
Crisp sets, 357
Cycles and multiple paths, 449

D

Data structures, 472
Decision problems, 141
Degree of vertices, 436
Depth – first search, 496
Derivations of grammar, 406
Derivation tree, 403
Deterministic finite-state automata, 418
Dfs algorithm, 499
Digraphs, 443
Di-orientable graph, 555
Directed cycles, 550
Directed paths, 547
Directed trees, 511
Disjunctive normal form, 141
Distributive inequalities, 284
Divisors of a given positive integer, 274
Domain, 239
Domain and range of a relation, 184
Don't care conditions, 334
Dual graphs, 530
Dual ideal, 291
Dual isomorphism, 280
Duality, 278

E

Edge colouring, 625
Efficient computer drum, 560
Electrical network problems, 470
Empty language, 397
Equality of boolean expressions, 319
Equality of two fuzzy sets, 360
Equality of words, 397
Equivalence relation, 441
Equivalence table, 152
Equivalent combinatorial circuits, 328
Eulerian circuit, 451
Eulerian circuit graph, 451
Eulerian graph, 450
Eulerian trail, 451
Euler's formula, 532
Exams scheduling problem, 617
Existential specification, 168

F

Factor-critical graph, 588
Final states, 414
Finite language, 396
Finite-state automation, 413, 419
Finite-state machine (FSM), 344, 394, 416
Five colour theorem, 633
Flashing red, 415
Floor and ceiling functions, 245
Floor function, 245
Flow of commodities problem, 445
Forest, 483
For example, 243
Formal languages, 396
Four colour problem, 467, 632
Free and bound variables, 166
Free monoid, 346
Free semigroup, 346
Free tree, 482
Functional completeness, 320
Functional digraphs, 490
Fusion of a graph, 458
Fuzziness, 364, 358
Fuzzy and control, 378
Fuzzy measures, 373

Fuzzy model, 382
Fuzzy relations, 369
Fuzzy sets, 357

G

Gallai and milgram theorem, 595
Games, 481
Geometrical representation of graphs, 522
Ghosts of departed graphs, 440
Ghouila – houri's theorem, 552
Grammar, 397, 403
Graph, 433
Graph of relations, 189
Graphs isomorphism, 453
Gray code, 198
Greatest integer function, 245
Greedy colouring, 631
Growth of functions, 262

H

Half-adder and full-adder, 326
Hamiltonian cycle, 451
Hamiltonian graph, 451
Handshaking lemma, 445
Hash address, 256
Hashing functions, 256
Hasse diagram, 275
Homeomorphic graph, 525
Huffman code, 518
Human operator, 384

I

Ideals of lattices, 291
Identity mapping, 244
Illustration of equivalence relations, 439
Image of a fuzzy set, 364
Implicant, 335
Implication table, 151
Improvable flows, 565
Induced subgraph, 450
Infant, adult, young and old fuzzy sets, 362
Infix, 517
Initial state, 413, 417
Injective mapping, 243
Inorder, 517
Input alphabet, 416
Input symbols, 414
Integer divisibility, 273
Integer value, 247
Intersection of fuzzy sets, 364
Intersection of two graphs, 455
Interval-valued fuzzy sets, 367
Into mappings, 243
Inverse image of a fuzzy sets, 364
Inverse mapping, 244
Inverse of a function, 250
Inverse of an I – V fuzzy relation, 372
Isolated vertex, 443
Isomorphism, 276
I–V fuzzy relation, 371

J

Job sequencing problem, 558
Joint and meet, 195

K

Karnaugh map, 331
K-flow, 568
Kleene closure, 399
Kleene operator, 399
Königsberg bridge problem, 467
König's theorem, 578, 627
K-outerplanar graphs, 536
Kruskal's algorithm, 504
Kuratowski's graphs, 526

L

Lattice homomorphism, 284, 287
Lattices as algebraic systems, 284
Leaf, 485
Least integer function, 245

Level, 517
Lexicographic order, 226, 274
Linear function, 244
List colouring, 630
List of implications, 151
Location graph, 435
Loops, 436

M

Many one mapping, 243
Marriage theorem, 579
Matching and covering in bipertite graphs, 577
Matching problem, 438
Max-flow min-cut theorem, 567
Maximum matching, 576
Membership function, 358
Minimal spanning trees, 504
Minimization process, 335
Minterms and maxterms, 319
Modular and distributive lattices, 292
Modular inequality, 284
Moon's and dirac's theorem, 550
Mtnl's networking problem, 470
Multiprocessor model, 395

N

NAND and NOR, 320
Network flows, 564
Next-state function, 414, 419
Nondeterministic turing machine, 395
Nonterminal, 404
Normal fuzzy set, 361, 389
Null graph, 442
Number of distinct relation from set A to B, 185

O

Onto mapping, 243
Operations of fuzzy sets, 362
Optimal assignment problem, 601
Oracle turing machine, 395
Order isomorphism, 284
Outerplanar graphs, 535
Output function, 417
Output symbols, 416

P

Parallel models, 395
Parallel random-access machine, 396
Parent, 485
Parse tree, 406
Parsing, 406
Partial ordered sets, 272
Partial order relation, 225
Partitions, 182
Path, 446
Path covers, 595
Paths in relations and digraphs, 191
Pendent vertex, 442
Perfect match, 438
Perfect matching, 584
Permutation function, 261
Phase-structure grammar, 406
Planar graph, 521
Pointer machine, 395
Polynomial function, 244
Polynomial of degree, 244
Postfix, 517
Postorder traversal, 517
Pragmatic, 375
Prefect matching, 576
Prefix, 398, 517
Preorder, 517
Primitive recursive model, 394
Prim's algorithm, 509
Principal conjuctive normal forms, 148
Principal disjunctive normal form, 146
Probabilistic turing machine, 395
Probability measure, 374
Production rule, 404
Product of two posets, 280
Product sets, 182

Properties of inverse of a function, 250

Q

Quadratic function, 244
Quasi-strongly connected graph, 512

R

Ramsey theory, 637
Random access machine, 394
Random graph model, 442
Randomness, 364
Rank and nullity, 459
Ranking of the participants in a tournament, 562
Recursive clause, 319
Reflexive and irreflexive relations, 201
Regular language, 420
Relation as sets of ordered pairs, 190
Relations, 183
Remainder function, 248
Robbins theorem, 555
Rooted graph, 484
Rooted plane tree, 485
Routes between cities (geographical placement problem), 434
Rules of inference, 151

S

Satellite channel problem, 471
Scheduling of jobs, 636
Self dual, 278
Semantics, 375, 403
Sequential coloring algorithms, 631
Sequents, 153
Serial models, 394
Set inclusion, 273
Sex in america (matching problem), 437
Shannon's entropy, 384
Shared memory, 396
Shortest path, 572
Sibling, 485
Simple graph, 430, 444
Situations and puzzles, 481
Solution sets, 189
Spanning subgraph, 450
Spanning trees, 494
Special functions, 267
Special valid formulae using quantifiers, 167
Stanley theorem, 554
Start symbol, 404
Statement algebra, 273
Statement functions, 166
States, 413, 416
Storage problem, 637
String of formulae, 153
Subalgebraic structures, 274
Subgraph, 447
Sublattices, 291
Subproduct, 335
Substitution rule, 404
Subtree, 494
Suffix, 398
Sum and product of functions, 239
Sup-min composition of a relation with a fuzzy set, 370
Sup-min composition of two fuzzy relations, 370
Support of a fuzzy set, 361
Surjective mapping, 243
Switching algebra, 321
Symmetric, asymmetric and antisymmetric relations, 201
Symmetric binary relations, 444
Syntactic, 375
Syntax, 403, 405

T

Terminal symbol, 404
The goldner–harary graph, 533
The matrix of a relation, 190

Theory of inference for the predicate calculus, 168
The personnel assignment problem, 596
The predicate calculus, 164
The relation θ, 262
The social organization of sexuality, 437
The web, 472
Three utilities problem, 468
Time table problem, 635
Torus, 333
Totally ordered set, 226
Tournament, 549
Tournament ranking problem, 465
Traffic light, 415
Traffic model, 415
Trail, 446
Transcedental function, 247
Transition diagram, 417
Transition function, 414, 416
Transition table, 417
Traveling - salesman problem, 468
Tree, 481
Turing machine, 394
Tutte's problem, 569
Two queens puzzle, 254
Two-value possibility distributions, 388

U

Uncertainty and information, 375
Uncertainty and measures of fuzziness, 374
Union and intersection of two I–V fuzzy sets, 368
Union of fuzzy sets, 363
Union of two graphs, 455
Unique complement, 309
Unique identities, 309
Unique keys, 256
Unit delay machine, 423
Universal generalization, 168
Universal turing machine, 395
U-uncertainty, 386

V

Variables and quantifiers, 166
Vending machine, 416
Vertex colouring, 612
Vertex sequence, 446, 448
Vizing theorem, 628

W

Waiting-for-customer state, 416
Walk, 446
Warshall's algorithm, 216
Weighted adjacency matrix, 572
Weight in a graph, 495
Work-depth model, 395

Z

Zero ideal, 291